November 1988 LIDS-P-1832

Partially Asynchronous, Parallel Algorithms for Network Flow and Other Problems*

by

Paul Tseng†

Dimitri P. Bertsekas†

John N. Tsitsiklis†

Abstract

We consider the problem of computing a fixed point of a nonexpansive function f. We provide sufficient conditions under which a parallel, partially asynchronous implementation of the iteration $\mathbf{x} := f(\mathbf{x})$ converges. We then apply our results to (i) quadratic programming subject to box constraints (ii) strictly convex cost network flow optimization, (iii) an agreement and a Markov chain problem, (iv) neural network optimization, and (v) finding the least element of a polyhedral set determined by a weakly diagonally dominant, Leontief system. We finally present simulation results which illustrate the attainable speedup and the effects of asynchronism.

Key words: Parallel algorithms, asynchronous algorithms, nonexpansive functions, network flows, neural networks, agreement, Markov chains, Leontief systems.

---

* Work supported by the National Science Foundation under grants NSF-ECS-8519058 and NSF-ECS-8552419, and by the Army Research Office under grant DAAL03-86-K-0171.

† The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139.

Table of Contents

1. Introduction

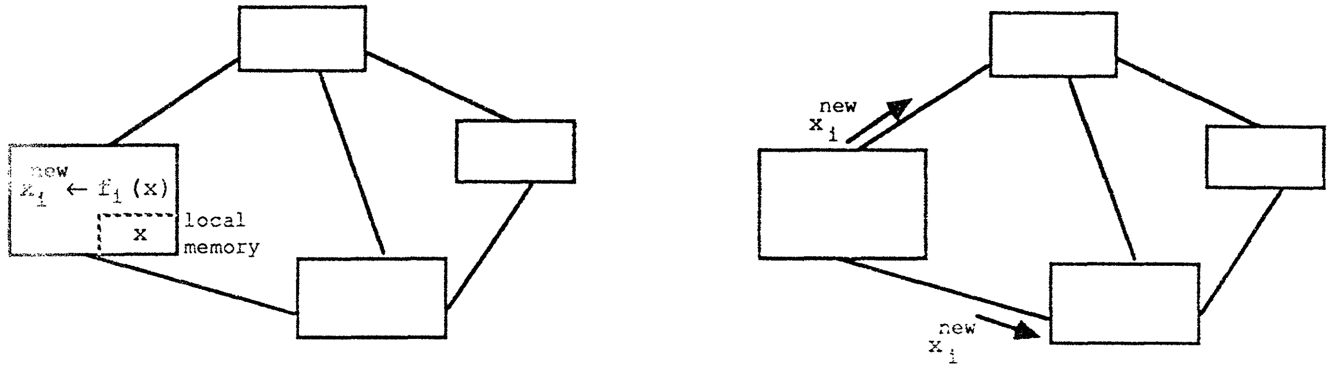
In this paper we consider the computation of a fixed point of a *nonexpansive* function f using parallel, partially asynchronous iterative algorithms of the form $x := f(x)$. We give sufficient conditions under which such algorithms converge, we show that some known methods [12], [15] are of this form, and we propose some new algorithms. The convergence behavior of the algorithms that we study is based on the nonexpansive property of the function f, and is qualitively different from the convergence behavior of most asynchronous algorithms that have been studied in the past [13]-[20].

We start by defining our model of asynchronous computation. The justification for this model and its interpretation can be found in [3]. Let $\Re^n$ denote the n-dimensional Euclidean space. For each $x \in \Re^n$, we denote by x_i the ith component of x, i.e., $x = (x_1,\ldots,x_n)$. We are given functions $f_i:\Re^n \to \Re$, $i=1,\ldots,n$, and we wish to find a point $x^* \in \Re^n$ such that

$$x^* = f(x^*),$$

where $f:\Re^n \to \Re^n$ is defined by $f(x) = (f_1(x),\ldots,f_n(x))$ (such a point x^* is called a *fixed point* of f).

We consider a network of processors endowed with local memories, which communicate by message passing, and which do not have access to a global clock. We assume that there are exactly n processors, each of which maintains its own estimate of a fixed point, and the ith processor is responsible for updating the ith component. (If the number of processors is smaller than n, we may let each processor update several components; the mathematical description of the algorithm does not change and our results apply to this case as well.) We assume that processor i updates its component by occasionally applying f_i to its current estimate, say x, and then transmitting (possibly with some delay) the computed value $f_i(x)$ to all other processors which use this value to update the ith component of their own estimates (see Figure 1.1).



Processor i computes new estimate of the ith component of a fixed point.

Processor i transmits new estimate to other processors.

Figure 1.1

We use a nonnegative integer variable t to index the events of interest (e.g. processor updates). We will refer to t as time, although t need not correspond to the time of a global clock. We use the notations:

$x_i(t)$ = ith component of the solution estimate known to processor i at time t.

$\mathcal{T}_i$ = an infinite set of times at which processor i updates x_i.

$\tau_{ij}(t)$ = a time at which the jth component of the solution estimate known to processor i at time t was in the local memory of processor j ($j = 1,\ldots,n$; $t \in \mathcal{T}_i$). [Naturally, $\tau_{ij}(t) \le t$.]

In accordance with the above definitions, we postulate that the variables $x_i(t)$ evolve according to:

$$x_i(t+1) = \begin{cases} f_i(x_1(\tau_{i1}(t)),\ldots,x_n(\tau_{in}(t))) & \text{if } t \in \mathcal{T}_i, \\ x_i(t) & \text{otherwise.} \end{cases} \tag{1.1}$$

A totally asynchronous model [16]-[20] is characterized by the assumption

$$\lim_{k\to+\infty} \tau_{ij}(t_k) = +\infty,$$

for all i, j and all sequences $\{t_k\} \subset T_i$ that tend to infinity. This assumption guarantees that given any time t_1, there exists some $t_2 > t_1$ such that $\tau_{ij}(t) \geq t_1$ for all i, j and $t \geq t_2$, that is, values of coordinates generated prior to t_1 will not be used in computations after a sufficiently large time t_2. On the other hand, the "delays" $t-\tau_{ij}(t)$ can become unbounded as t increases. This is the main difference with the following partial asynchronism assumption, where the amounts $t-\tau_{ij}(t)$ are assumed bounded (although the bound can be arbitrarily large).

Assumption A (Partial Asynchronism): There exists a positive integer B such that, for each i and each $t \in T_i$, there holds:

(a) $0 \leq t-\tau_{ij}(t) \leq B-1$, for all $j \in \{1,\ldots,n\}$.

(b) There exists $t' \in T_i$ for which $1 \leq t'-t \leq B$.

(c) $\tau_{ii}(t) = t$.

[For notational convenience we assume that initial conditions $x_i(1-B)$, $x_i(2-B)$, ..., $x_i(0)$ are given for each i, so that the asynchronous iteration (1.1) is well defined.] Parts (a) and (b) of Assumption A state that both the communication delays and the processor idle periods are bounded. Part (c) states that a processor never uses an outdated value of its own component. Assumption A can be seen to hold in many practical cases; for example, (b) holds if each processor uses a local clock, if the ratio of the speeds of different local clocks is bounded, and if each processor computes periodically according to its own local clock.

Partially asynchronous iterations have already been studied in the context of gradient optimization algorithms, for which it was shown that convergence is obtained provided that the bound B of Assumption A is sufficiently small [13]-[15]. Our results concern a different class of methods for which convergence is established for every value of the bound B. This may be somewhat surprising in view of the fact that the totally asynchronous versions of the methods considered here do not converge in general. Results of this type are known so far only for an "agreement" [15] and a Markov chain [12] problem (which are revisited in §5). Our main result (Proposition 2.1) is the first general convergence result for methods exhibiting this particular convergence behavior. It is then shown in subsequent sections that Proposition 2.1 applies to a variety of methods for several important problems. In fact, some of our convergence results are new even when they are specialized to the case of synchronous algorithms, that is, algorithms in which no delays are present (B = 1).

2. A General Convergence Theorem

Throughout this paper, we use the following notations:

$$
\begin{aligned}
X^* &= \{ x \mid x\in\Re^n,\ f(x)=x \},\\
N &= \{1,\ldots,n\},\\
|S| &= \text{cardinality of } S, \text{ where } S \text{ is a finite set},\\
\|x\| &= \max\{ |x_i| \mid i\in N \}, \text{ where } x\in\Re^n,\\
\rho(x,X) &= \inf\{ \|x-y\| \mid y\in X \}, \text{ where } x\in\Re^n \text{ and } X\subseteq\Re^n \text{ is nonempty},\\
\Omega &= \{ (x^*,\beta,S^-,S^+) \mid x^*\in X^*,\ \beta\in(0,+\infty),\ S^-\subseteq N,\ S^+\subseteq N,\\
&\qquad S^-\cup S^+\neq\emptyset,\ S^-\cap S^+=\emptyset \}.
\end{aligned}
$$

Also, for each $(x^*,\beta,S^-,S^+)\in\Omega$, we denote

$$
\begin{aligned}
F(x^*,\beta,S^-,S^+) = \{ x \mid\ & x_i^*-x_i=\beta,\ \forall\, i\in S^-,\ x_i-x_i^*=\beta,\ \forall\, i\in S^+,\\
& |x_i-x_i^*|<\beta,\ \forall\, i\notin (S^-\cup S^+) \}.
\end{aligned}
$$

In words, $F(x^*,\beta,S^-,S^+)$ is the relative interior of an $(n-|S^-\cup S^+|)$-dimensional face of the cube in $\Re^n$ centered at x^* with edge length 2β (the sets S^- and S^+ serve to specify a particular face). Our main assumption on the structure of f is the following:

Assumption B

(a) f is continuous.

(b) X^* is convex and nonempty.

(c) $\|f(x)-x^*\| \le \|x-x^*\|$, $\forall$ $x \in \Re^n$, $\forall$ $x^* \in X^*$.

(d) For any $(x^*,\beta,S^-,S^+) \in \Omega$, there exists $s \in (S^- \cup S^+)$ such that $f_s(x) \ne x_s$ for all $x \in F(x^*,\beta,S^-,S^+)$ with $\rho(x,X^*) = \beta$.

Part (c) of Assumption B states that f does not increase the distance from a fixed point and will be referred to as the pseudo-nonexpansive property. This is slightly weaker than requiring that f be nonexpansive (that is, $\|f(x)-f(y)\| \le \|x-y\|$ for all x and y in $\Re^n$) and in certain cases may be easier to verify. Part (d) states that for any $x^* \in X^*$ and $\beta > 0$, if F is the relative interior of a face of the n-dimensional cube around x^* with edge length 2β, then f moves the same component x_s of all points in F for which x^* is the nearest element of X^* (see Figure 2.1). Furthermore s is a "worst" component index, in the sense that $|x_s-x_s^*| = \beta = \|x-x^*\|$. This part of Assumption B is usually the most difficult to verify in specific applications.

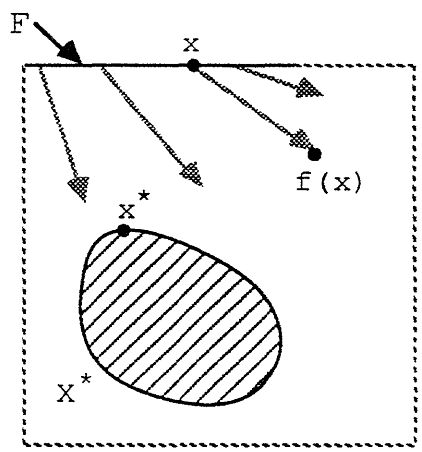


Figure 2.1. f moves the same component of all points in F for which x^* is the nearest element of X^*.

The convexity of X^* is also sometimes hard to verify. For this reason we will consider another assumption that is stronger than Assumption B, but is easier to verify.

Assumption B′

(a) f is continuous.

(b) X^* is nonempty.

(c) $\|f(x)-x^*\| \leq \|x-x^*\|$, $\forall\ x \in \Re^n$, $\forall\ x^* \in X^*$.

(d) For any $(x^*,\beta,S^-,S^+) \in \Omega$, there exists $s \in (S^- \cup S^+)$ such that $f_s(x) \neq x_s$ for all $x \in F(x^*,\beta,S^-,S^+)$ with $x \notin X^*$.

Compared to Assumption B, part (d) of the new assumption is stronger but part (b) is weaker because convexity is not assumed. We have the following result:

Lemma 2.1 Assumption B′ implies Assumption B.

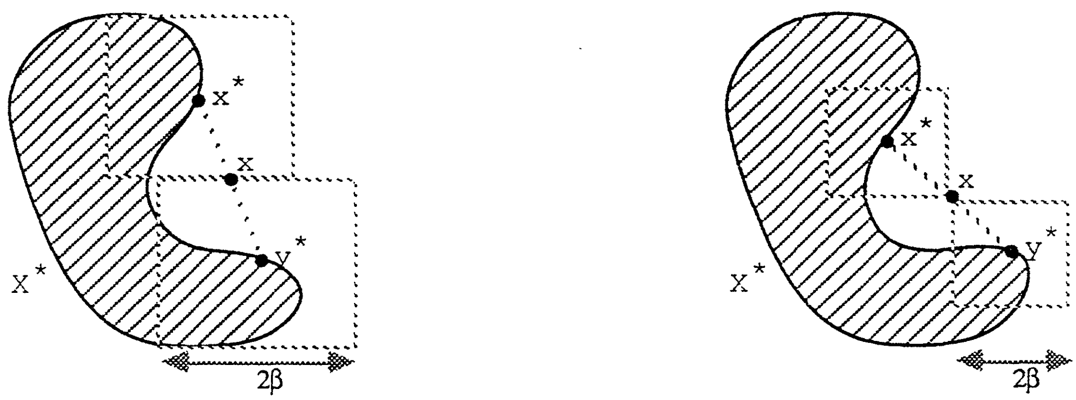
Proof: We only need to show that X^* is convex. Suppose the contrary. Then there exist $x^* \in X^*$ and $y^* \in X^*$ such that $(x^*+y^*)/2 \notin X^*$. Let

$$\beta = \|x^*-y^*\|/2,\quad S^1 = \{\, i \mid x_i^*-y_i^* = 2\beta \,\},\quad S^2 = \{\, i \mid y_i^*-x_i^* = 2\beta \,\}$$

and $x = (x^*+y^*)/2$. Then $(x^*,\beta,S^1,S^2) \in \Omega$, $x \notin X^*$ and

$$x \in F(x^*,\beta,S^1,S^2),\quad x \in F(y^*,\beta,S^2,S^1)$$

(see Figure 2.2).



$S^1 = \{2\}$, $S^2 = \emptyset$. $S^1 = \{2\}$, $S^2 = \{1\}$.

Figure 2.2. Two examples of x^*, y^*, S^1 and S^2.

By assumption there exists $s \in (S^1 \cup S^2)$ such that $f_s(x) \neq x_s$. Suppose that $s \in S^1$. Then if $f_s(x) > x_s$, we obtain $\|f(x) - y^*\| > \beta$ and if $f_s(x) < x_s$, we obtain $\|f(x) - x^*\| > \beta$. In either case Assumption B′ (c) is contradicted. The case where $s \in S^2$ is treated analogously. Q.E.D.

Assumption B will be used in §4, while Assumption B′ will be used in §3, §6 and §7. Unfortunately, the following simple example shows that Assumptions A and B alone are not sufficient for convergence of the asynchronous iteration (1.1): Suppose that $f(x_1, x_2) = (x_2, x_1)$ (which can be verified to satisfy Assumption B with $X^* = \{(\lambda, \lambda) \mid \lambda \in \Re\}$). Then the sequence $\{x(t)\}$ generated by the synchronous iteration $x(t+1) = f(x(t))$ (which is a special case of (1.1)), with $x(0) = (1,0)$, oscillates between $(1,0)$ and $(0,1)$. To prevent such behavior, we introduce an additional assumption:

<u>Assumption C</u>: For any $x \in \Re^n$, any $x^* \in X^*$, and any $s \in N$, if $f_s(x) \neq x_s$ then $|f_s(x) - x_s^*| < \|x - x^*\|$.

We show below that Assumption C can be enforced by introducing a relaxation parameter:

<u>Lemma 2.2</u> Let $h: \Re^n \to \Re^n$ be a function satisfying Assumption B. Then the mapping $f: \Re^n \to \Re^n$ whose ith component is

$$f_i(x) = (1 - \gamma_i) x_i + \gamma_i h_i(x),$$

where $\gamma_1, \ldots, \gamma_n$ are scalars in $(0,1)$, has the same set of fixed points as h and satisfies both Assumptions B and C.

<u>Proof</u>: It is easily seen that f is continuous and has the same set of fixed points as h. This, together with the observation that $f_i(x) \neq x_i$

if and only if $h_i(x) \neq x_i$, implies that f satisfies parts (a), (b) and (d) of Assumption B. For any $x \in \Re^n$ and any $x^* \in X^*$, we have, using the pseudo-nonexpansive property of h,

$$\begin{aligned} \|f(x)-x^*\| &= \|(1-\gamma_i)(x-x^*)+\gamma_i(h(x)-x^*)\| \\ &\leq (1-\gamma_i)\|x-x^*\|+\gamma_i\|x-x^*\| \\ &= \|x-x^*\|, \end{aligned}$$

and therefore f also satisfies part (c) of Assumption B.

It remains to show that f satisfies Assumption C. Consider any $x \in \Re^n$, any $x^* \in X^*$, and any $s \in N$ such that $f_s(x) \neq x_s$. Then $h_s(x) \neq x_s$. This, together with the fact that both x_s and $h_s(x)$ belong to the interval $[x_s^*-\|x-x^*\|, x_s^*+\|x-x^*\|]$ and $\gamma_s \in (0,1)$, implies that $f_s(x) = (1-\gamma_s)x_s+\gamma_s h_s(x)$ is in the <u>interior</u> of the same interval. Q.E.D.

We will show next that Assumptions A, B and C are sufficient for the sequence $\{x(t)\}$ generated by (1.1) to converge to an element of X^*. To motivate our proof, consider the synchronous iteration $x(t+1) = f(x(t))$. Under Assumptions B and C, either (i) $\rho(x(t+1),X^*) < \rho(x(t),X^*)$ or (ii) $\rho(x(t+1),X^*) = \rho(x(t),X^*)$ and $x(t+1)$ has a smaller number of components at distance of $\rho(x(t),X^*)$ from X^* than $x(t)$. Thus case (ii) can occur for at most n successive iterations before case (i) occurs. For the partially asynchronous iteration (1.1), because of communication and computation delays (each bounded by B, due to Assumption A), the number of time steps until the distance to X^* decreases is upper bounded by roughly 2nB.

<u>Proposition 2.1</u> Suppose that $f:\Re^n \rightarrow \Re^n$ satisfies Assumptions B and C, and suppose that Assumption A (partial asynchronism) holds. Then the sequence $\{x(t)\}$ generated by the asynchronous iteration (1.1) converges to some element of X^*.

Proof: For each $t \geq 0$ denote

$$z(t) = (x(t-B+1), \ldots, x(t)),$$

$$d(z(t)) = \inf_{x^* \in X^*}\{ \max\{\|x(t-B+1)-x^*\|, \ldots, \|x(t)-x^*\|\} \}.$$

Notice that the infimum in the definition of $d(z(t))$ is attained because the set X^* is closed (as a consequence of the continuity of f).

Lemma 2.3

(a) $d(z(t+1)) \leq d(z(t)), \ \forall z(t) \in \Re^{nB}$.

(b) If $d(z(t)) > 0$, then $d(z(t+2nB+B-1)) < d(z(t)), \ \forall z(t) \in \Re^{nB}$.

Proof: Let $\beta = d(z(t))$ and let x^* be an element of X^* for which

$$\max\{\|x(t-B+1)-x^*\|, \ldots, \|x(t)-x^*\|\} = \beta. \tag{2.1}$$

First we claim that

$$\|x(r)-x^*\| \leq \beta, \ \forall r \geq t-B+1. \tag{2.2}$$

From (2.1), this claim holds for $r \in \{t-B+1, \ldots, t\}$. Suppose that it holds for all $r \in \{t-B+1, \ldots, r'\}$, where r' is some integer greater than or equal to t, and we will show that it holds for $r'+1$. By (1.1), for each $i \in N$, $x_i(r'+1)$ equals either $x_i(r')$ or $f_i(x_1(\tau_{i1}(r')), \ldots, x_n(\tau_{in}(r')))$. In either case we have (cf. Assumption A (a) and Assumption B (c)) that

$$|x_i(r'+1)-x_i^*| \leq \max\{\|x(r'-B+1)-x^*\|, \ldots, \|x(r')-x^*\|\} \leq \beta,$$

where the last inequality follows from the induction hypothesis. This proves (2.2). Eqs. (2.1) and (2.2) immediately imply (a).

We now prove (b). Suppose that $d(z(t)) > 0$. With x^* and β defined as before (cf. Eq. (2.1)), denote

$$S^-(r) = \{ i \mid x_i(r)-x_i^* = -\beta \}, \ S^+(r) = \{ i \mid x_i(r)-x_i^* = \beta \}.$$

First we claim that

$$S^-(r+1) \subseteq S^-(r), \ S^+(r+1) \subseteq S^+(r), \ \forall r \geq t. \tag{2.3}$$

To see this, fix any $r \geq t$ and let i be any element of $N\backslash S^-(r)$.† Then (cf. (2.2))

$$x_i(r) - x_i^* > -\beta. \tag{2.4}$$

By (1.1), either (i) $x_i(r+1) = x_i(r)$ or (ii) $x_i(r+1) =$

$f_i(x_1(\tau_{i1}(r)),\ldots,x_n(\tau_{in}(r))) \neq x_i(r) = x_i(\tau_{ii}(r))$. (The last equality follows from Assumption A (c).) In case (i), (2.4) implies that $x_i(r+1) - x_i^* > -\beta$. In case (ii), Assumption C implies that

$$|x_i(r+1) - x_i^*| < \max_j |x_j(\tau_{ij}(r)) - x_j^*| \leq \beta,$$

where the last inequality follows from (2.2). Therefore in either case $i \in N\backslash S^-(r+1)$. By an analogous argument we can show that $i \in N\backslash S^+(r)$ implies $i \in N\backslash S^+(r+1)$, and (2.3) is proven.

Let $S(r) = S^+(r) \cup S^-(r)$ for all r. We next claim that, for $r \geq t$,

$$d(z(r+2B)) = \beta \quad \Rightarrow \quad S(r+2B) \neq S(r). \tag{2.5}$$

To show this we will argue by contradiction. Suppose that, for some $r \geq t$, we have $d(z(r+2B)) = \beta$ and $S(r) = S(r+2B)$. By (2.3), $S^-(r) = S^-(r+1) = \ldots = S^-(r+2B)$ and $S^+(r) = S^+(r+1) = \ldots = S^+(r+2B)$. Let $S = S(r)$, $S^- = S^-(r)$ and $S^+ = S^+(r)$. Then (cf. (2.2))

$$x_i(r) = x_i(r+1) = \ldots = x_i(r+2B) = x_i^* - \beta, \ \forall\, i \in S^-, \tag{2.6}$$

$$x_i(r) = x_i(r+1) = \ldots = x_i(r+2B) = x_i^* + \beta, \ \forall\, i \in S^+, \tag{2.7}$$

$$|x_i(r) - x_i^*| < \beta, \ \ldots, \ |x_i(r+2B) - x_i^*| < \beta, \ \forall\, i \notin S. \tag{2.8}$$

By Assumption A (b) and (1.1), for each $i \in S$, there exists $r_i \in \{r+B,\ldots,r+2B-1\}$ such that

$$x_i(r_i+1) = f_i(x_1(\tau_{i1}(r_i)),\ldots,x_n(\tau_{in}(r_i))).$$

This, together with (2.6)-(2.8) and Assumption A (a), implies that

$$x_i^i = f_i(x^i), \ x^i \in F(x^*,\beta,S^-,S^+), \ \forall\, i \in S,$$

where $x^i = (x_1(\tau_{i1}(r_i)),\ldots,x_n(\tau_{in}(r_i)))$. By Assumption B (d), there

---

† The notation $A\backslash B$ stands for $A \cap \overline{B}$, where $\overline{B}$ is the complement of B.

exists $s \in S$ for which $\rho(x^s, X^*) < \beta$, i.e., there exist $y^* \in X^*$ and $\theta \in [0, \beta)$ such that $\|x^s - y^*\| = \theta$. Let

$$\varepsilon = \max\{\ |x_i(m) - x_i^*|\ \mid\ i \notin S,\ m = r+B, \ldots, r+2B-1\ \},$$

$$M = \max\{\ |x_i(m) - y_i^*|\ \mid\ i \notin S,\ m = r+B, \ldots, r+2B-1\ \}$$

(see Figure 2.3).

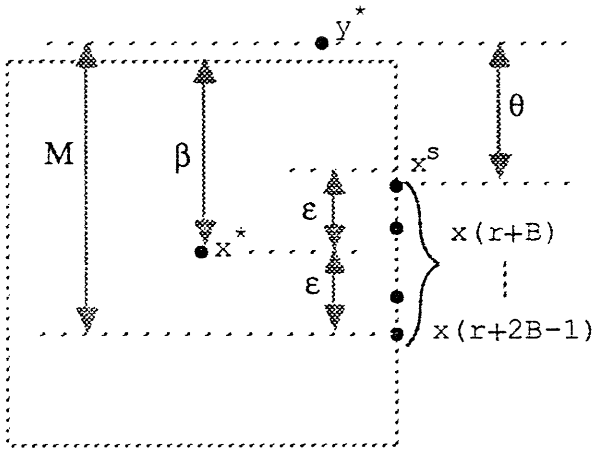


Figure 2.3

Since X^* is convex, we have that, for any $\omega \in (0,1)$, $z^* = (1-\omega)x^* + \omega y^*$ is in X^* and, for $m = r+B, \ldots, r+2B-1$,

$$|x_i(m) - z_i^*| = |x_i^s - z_i^*|$$

$$\leq (1-\omega)|x_i^s - x_i^*| + \omega|x_i^s - y_i^*|$$

$$= (1-\omega)\beta + \omega\theta, \quad \forall\ i \in S,$$

$$|x_i(m) - z_i^*| \leq (1-\omega)|x_i(m) - x_i^*| + \omega|x_i(m) - y_i^*|$$

$$\leq (1-\omega)\varepsilon + \omega M, \quad \forall\ i \notin S.$$

Since $\varepsilon < \beta$ and $\theta < \beta$, we have that, for ω sufficiently small,

$$\|x(r+B) - z^*\| < \beta,\ \ldots\ , \|x(r+2B-1) - z^*\| < \beta.$$

This implies that $d(z(r+2B-1)) < \beta$, a contradiction.

Lemma 2.3 (a) and Eq. (2.5) imply that either $d(z(t+2nB-1)) < \beta$ or (cf. (2.3)) $S(t+2nB) = \ldots = S(t+2nB+B-1) = \varnothing$. In the latter case we obtain that

$$d(z(t+2nB+B-1)) = \max\{\|x(t+2nB)-x^*\|,\ldots,\|x(t+2nB+B-1)-x^*\|\} < \beta.$$

Q.E.D.

By Lemma 2.3, the sequence $\{z(t)\}$ is bounded and $d(z(t))$ monotonically decreases to some limit β. If $\beta = 0$, then (cf. (2.2)) $\{z(t)\}$ has a unique limit point (which is in X^*) and our proof is complete. Suppose that $\beta > 0$, and we will obtain a contradiction. Let

$$t^* = 2nB+B-1. \tag{2.9}$$

Since $\{z(t)\}$ is bounded, there exists some $z^* \in \Re^{nB}$, $z^{**} \in \Re^{nB}$ and subsequence T of $\{0,1,\ldots\}$ such that

$$\{z(t)\}_{t\in T} \to z^*, \quad \{z(t+t^*)\}_{t\in T} \to z^{**}. \tag{2.10}$$

Note that since $d(z(t)) \to \beta$ and d is a continuous function, (2.10) implies that $d(z^*) = d(z^{**}) = \beta$.

From (1.1), Assumption A and the definition of $z(t)$, we see that, for all $t \geq 0$,

$$z(t+t^*) = u(z(t);\theta(t)), \tag{2.11}$$

where $\theta(t)$ denotes the set

$$\{\ (r-t,\tau_{i1}(r)-t,\ldots,\tau_{in}(r)-t)\ \mid\ r\in T^i \cap \{t,\ldots,t+t^*\},\ i\in N\ \} \tag{2.12}$$

and $u(\cdot;\theta(t)):\Re^{nB}\to\Re^{nB}$ is some continuous function that depends on f and $\theta(t)$ only. (Note that $u(\cdot;\theta(t))$ is the composition of the f_i's in an order determined by $\theta(t)$ and is continuous because f is continuous). Since (cf. (2.12) and Assumption A) $\theta(t)$ takes values from a <u>finite</u> set, by further passing into a subsequence if necessary, we can assume that $\theta(t)$ is the same set for all $t\in T$. Let θ denote this set. Then from (2.11) we obtain that

$$z(t+t^*) = u(z(t);\theta), \ \forall\, t\in T.$$

Since $u(\cdot;\theta)$ is continuous, this together with (2.10) implies that $z^{**} =$

$u(z^*;\theta)$ or, equivalently, $z(t^*) = z^{**}$ if $z(0) = z^*$ and

$$\{ (r,\tau_{i1}(r),\ldots,\tau_{in}(r)) \mid r\in T^i\cap\{0,\ldots,t^*\},\ i\in N \} = \theta.$$

Since $d(z^*) = \beta > 0$, this together with Lemma 2.3 (b) implies that $d(z^{**}) < d(z^*)$ - contradicting the hypothesis $d(z^{**}) = \beta$. Q.E.D.

Proposition 2.1 can be easily generalized in a number of directions:

a) Replace the maximum norm by a weighted maximum norm of the form $\|x\| = \max_i |x_i|/w_i$, where w_i is a positive scalar for all i.

b) Consider functions $f:X\to X$, where $X = I_1\times\ldots\times I_n$ and each I_i is a closed interval in $\Re$.

c) Consider time dependent iterations of the form $x(t+1) := f(t,x(t))$.

It should be stressed that convergence fails to hold if Assumption A is replaced by the total asynchronism assumption

$$\lim_{t\to+\infty} \tau_{ij}(t) = +\infty,\ \forall\ i,\ j.$$

Some divergent examples can be found in Section 7.1 of [3].

3. Nonexpansive Mappings on a Box

Let $g:\Re^n \to \Re^n$ be a continuously differentiable function satisfying the following assumption:

Assumption D

(a) For each $i \in N$, $\sum_{j\in N} |\partial g_i(x)/\partial x_j| \le 1$, $\forall\ x \in \Re^n$.

(b) For each $i \in N$ and $j \in N$, either $\partial g_i(x)/\partial x_j = 0$, $\forall\ x \in \Re^n$ or $\partial g_i(x)/\partial x_j \ne 0$, $\forall\ x \in \Re^n$.

(c) The graph $(N, \{ (i,j) \mid \partial g_i(x)/\partial x_j \ne 0 \})$ is strongly connected.

Let C be a box (possibly unbounded) in $\Re^n$, i.e.,

$$C = \{ x \in \Re^n \mid l_i \le x_i \le c_i\ \forall\ i \in N \},$$

for some scalars l_i and c_i satisfying $l_i \le c_i$ (we allow $l_i = -\infty$ or $c_i = +\infty$), and let $[\,]^+$ denote the orthogonal projection onto C, i.e., $[x]^+ = (\max\{l_1,\min\{c_1,x_1\}\},\ldots,\max\{l_n,\min\{c_n,x_n\}\})$. We use the notation x^T to denote the transpose of a vector x. The following is the main result of this section:

Proposition 3.1 Let $g:\Re^n \to \Re^n$ satisfy Assumption D. If either g has a fixed point or if C is bounded, then the function $h:\Re^n \to \Re^n$ defined by

$$h(x) = [g(x)]^+ \tag{3.1}$$

satisfies Assumption B′.

Proof: Since both g and $[\,]^+$ are continuous functions, so is their composition and part (a) of Assumption B′ holds.

Consider any $i \in N$. By the Mean Value Theorem, for any $x \in \Re^n$ and any $y \in \Re^n$, there exists $\xi \in \Re^n$ such that

$$g_i(y) - g_i(x) = (\nabla g_i(\xi))^T (y-x). \tag{3.2}$$

This implies that

$$|g_i(y)-g_i(x)| \le \sum_j |\partial g_i(\xi)/\partial x_j|\,|y_i-x_i|$$
$$\le (\sum_j |\partial g_i(\xi)/\partial x_j|)\cdot\|x-y\|$$
$$\le \|x-y\|,$$

where the last inequality follows from Assumption D (a). Since the choice of i was arbitrary, g is nonexpansive. Since $[\,]^+$ is easily seen to be nonexpansive, it follows that $\|h(x)-h(y)\| \le \|g(x)-g(y)\|$. Thus, h is nonexpansive and part (c) of Assumption B′ is satisfied.

We now show that h has a fixed point. Suppose first that g has a fixed point y^*. Choose β sufficiently large so that the set $Y = \{ x\in\Re^n \mid \|x-y^*\| \le \beta \}\cap C$ is nonempty. Then for every $x\in Y$ we have, for all i,

$$y_i^*-\beta \le g_i(x) \le y_i^*+\beta,$$

and

$$\text{either} \quad l_i \le g_i(x) \le c_i \quad \text{or} \quad g_i(x) < l_i \le y_i^*+\beta \quad \text{or} \quad y_i^*-\beta \le c_i < g_i(x).$$

Since $h_i(x) = \max\{l_i,\min\{c_i,g_i(x)\}\}$, this implies that $h(x)\in Y$ (see Figure 3.1 below).

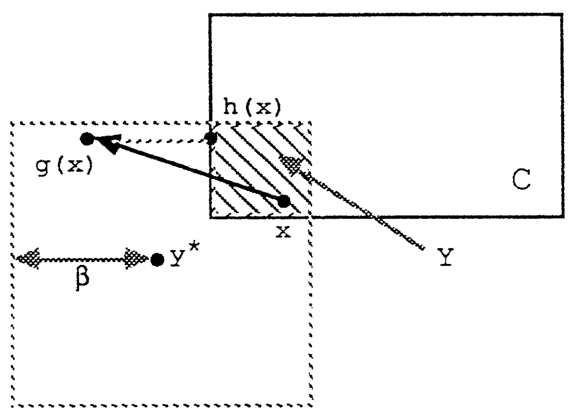


Figure 3.1

Since h is also continuous and Y is convex and compact, a theorem of Brouwer ([6], pp. 17) shows that h has a fixed point. Now suppose C is bounded. Since $h(x)\in C$ for all $x\in C$ and C is convex and compact, the same theorem of Brouwer shows that h has a fixed point.

We will show that Assumption B′ (d) holds. Suppose the contrary.

Then there exists some $(x^*, \beta, S^-, S^+) \in \Omega$ such that for every $s \in (S^- \cup S^+)$ there is a $x^s \in F(x^*, \beta, S^-, S^+)$ such that $x^s \notin X^*$ and $h_s(x^s) = x_s{}^s$. Let $S = S^- \cup S^+$ and fix some $s \in S$. By the Mean Value Theorem, there exists some $\xi \in \Re^n$ such that $g_s(x^s) - g_s(x^*) = (\nabla g_s(\xi))^T (x^s - x^*)$. Let $a_j = \partial g_s(\xi)/\partial x_j$. Then

$$\begin{aligned}\beta &= |x_s{}^s - x_s{}^*| = |h_s(x^s) - h_s(x^*)| \\ &\le |g_s(x^s) - g_s(x^*)| \\ &= |\textstyle\sum_j a_j(x_j{}^s - x_j{}^*)| \\ &\le (\textstyle\sum_{j\in S} |a_j|)\beta + (\textstyle\sum_{j\notin S} |a_j|\,|x_j{}^s - x_j{}^*|) \\ &\le \beta + \textstyle\sum_{j\notin S} |a_j|(|x_j{}^s - x_j{}^*| - \beta),\end{aligned}$$

where the first inequality follows from the fact that the projection onto $[l_s, c_s]$ is nonexpansive and the last inequality follows from the fact (cf. Assumption D (a)) that $\sum_j |a_j| \le 1$. Since $|x_j{}^s - x_j{}^*| < \beta$ for all $j \notin S$, the above inequality implies that $a_j = 0$ for all $j \notin S$. Since the choice of $s \in S$ was arbitray, we obtain from Assumption D (b) that $\partial g_s(x)/\partial x_j = 0$ for all $x \in \Re^n$, $s \in S$, $j \notin S$. By Assumption D (c), it must be that $S = N$. In that case $F(x^*, \beta, S^+, S^-)$ is a singleton and all the vectors x^s are equal. The equalities $h_s(x^s) = x^s$, for all s, imply that each x^s is a fixed point of h - a contradiction of the hypothesis $x^s \notin X^*$. Q.E.D.

Since Assumption B′ is satisfied, the partially asynchronous iteration

$$x := (1-\gamma)x + \gamma[g(x)]^+$$

(with $0 < \gamma < 1$) converges, (Lemmas 2.1, 2.2 and Proposition 2.1). An important special case is obtained if $C = \Re^n$, $g(x) = Ax+b$, where A is an n×n matrix and b is a given vector in $\Re^n$. Assumption D amounts to the requirement that $A = [a_{ij}]$ is irreducible (see [9] for a definition of irreducibility) and $\sum_j |a_{ij}| \le 1$, for all i. Then, provided that the system $x = Ax+b$ has a solution, the partially asynchronous iteration

$$x := (1-\gamma)x+\gamma(Ax+b)$$

converges to it (with $0 < \gamma < 1$). As a special case of our results we obtain convergence of the synchronous iteration

$$x(t+1) = (1-\gamma)x(t)+\gamma(Ax(t)+b).$$

This seems to be a new result under our assumptions. Previous convergence results [9], [27] have made the stronger assumption that either (a) A is irreducible and $\sum_j |a_{ij}| \le 1$, for all i, with strict inequality for at least one i, or (b) $\sum_j |a_{ij}| < 1$, for all i. Two other important special cases are studied below.

3.1 Quadratic Costs Subject to Box Constraints

Consider the following problem

$$\text{Minimize} \quad x^T Qx/2 + q^T x \qquad (3.3)$$

$$\text{Subject to} \quad x \in C,$$

where $Q = [q_{ij}]$ is a symmetric, irreducible, nonnegative definite matrix of dimension $n \times n$ satisfying

$$\sum_{j \ne i} |q_{ij}| \le q_{ii},\ q_{ii} > 0,\ \forall\ i \in N, \qquad (3.4)$$

q is an element of $\Re^n$, and C is, as before, a box in $\Re^n$.

Let D denote the diagonal matrix whose ith diagonal entry is q_{ii}. Let $A = I-D^{-1}Q$ and $b = -D^{-1}q$. We have the following:

Proposition 3.2 The function $g:\Re^n \to \Re^n$ defined by $g(x) = Ax+b$ satisfies Assumption D.

Proof: g is clearly continuously differentiable and (cf. (3.4)) $\sum_j |a_{ij}| = \sum_{j \neq i} |q_{ij}|/q_{ii} \leq 1$ for all i. Since $\partial g_i(x)/\partial x_j = a_{ij}$ for all $x \in \Re^n$ and A is irreducible, g satisfies Assumption D. Q.E.D.

It can be seen (using the Kuhn-Tucker optimality conditions [10]) that each optimal solution of (3.3) is a fixed point of $[Ax+b]^+$ and vice versa, where $[\,]^+$ denotes the orthogonal projection onto C. Hence, if (3.3) has an optimal solution, then (cf. Lemma 2.2 and Propositions 2.1, 3.1, 3.2) the partially asynchronous implementation of the iteration

$$x := (1-\gamma)x + \gamma[Ax+b]^+,$$

where $\gamma \in (0,1)$, converges to a solution of (3.3).

3.2. Separable Quadratic Costs with Sparse 0,+1,-1 Matrix

Consider the following problem

$$\begin{aligned} \text{Minimize} \quad & w^T D w/2 + \beta^T w \\ \text{Subject to} \quad & Ew \geq d, \end{aligned} \tag{3.5}$$

where D is an m×m positive-definite diagonal matrix, β is an element of $\Re^m$, d is an element of $\Re^n$, and $E = [e_{ik}]$ is an n×m matrix having at most two nonzero entries per column and each nonzero entry is either -1 or 1. Furthermore, we assume that the undirected graph $\mathcal{G}$ with nodes $\{1,\ldots,n\}$ and arcs $\{(i,j) \mid e_{ik} \neq 0 \text{ and } e_{jk} \neq 0 \text{ for some } k\}$ is connected.

Consider the following Lagrangian dual [10] of (3.5)

$$\begin{aligned} &\text{Minimize} \quad x^TQx/2 + q^Tx \\ &\text{Subject to} \quad x \geq 0, \end{aligned} \tag{3.6}$$

where $Q = ED^{-1}E^T$, $q = -d-ED^{-1}\beta$. We show below that this is a special case of the problem considered in the previous subsection.

Proposition 3.3 Q is symmetric, irreducible, nonnegative definite and satisfies (3.4).

Proof: Since D is symmetric and positive definite, Q is symmetric and nonnegative definite. To see that Q satisfies (3.4), let α_k denote the kth diagonal entry of D ($\alpha_k > 0$), let $O(i)$ denote the set of indexes k such that $e_{ik} \neq 0$, and let q_{ij} denote the (i,j)th entry of Q. Then

$$\begin{aligned} |q_{ij}| &= |\textstyle\sum_k e_{ik}(\alpha_k)^{-1}e_{jk}| \\ &\leq \textstyle\sum_{k\in O(i)\cap O(j)} (\alpha_k)^{-1}, \end{aligned}$$

with equality holding if $i = j$. Hence, for each i,

$$\begin{aligned} \textstyle\sum_{j\neq i}|q_{ij}| &\leq \textstyle\sum_{j\neq i} \textstyle\sum_{k\in O(i)\cap O(j)} (\alpha_k)^{-1} \\ &\leq \textstyle\sum_{k\in O(i)} (\alpha_k)^{-1} \\ &= q_{ii}, \end{aligned}$$

where the second inequality follows from the fact that if $k\in O(i)\cap O(j)$ for some j, then $k\notin O(i)\cap O(j')$ for all j' not equal to i or j. Finally, Q is irreducible because $\mathcal{G}$ is connected and $q_{ij} \neq 0$ for $i \neq j$ if and only if there exists some k such that $e_{ik} \neq 0$ and $e_{jk} \neq 0$. Q.E.D.

An example of constraints $Ew \geq d$ satisfying our conditions on E is

$$\textstyle\sum_k w_k \leq 1 \quad \text{and} \quad \textstyle\sum_{k\in K_r} w_k \geq 0 \text{ for } r = 1,2,\ldots,R,$$

where $K_1, K_2, \ldots, K_R$ are some mutually disjoint subsets of $\{1,2,\ldots,m\}$. Such constraints often arise in resource allocation problems.

4. Strictly Convex Cost Network Flow Problems

Let $(\mathcal{N},\mathcal{A})$ be a connected, directed graph (network), where $\mathcal{N} = \{1,\ldots,n\}$ is the set of nodes and $\mathcal{A}\subseteq\mathcal{N}\times\mathcal{N}$ is the set of arcs. We assume that $i \neq j$ for every arc (i,j). For each $i\in\mathcal{N}$, denote by $\mathcal{D}(i)$ the set of downstream neighbors of i (that is, $\mathcal{D}(i) = \{ j \mid (i,j)\in\mathcal{A} \}$) and by $\mathcal{U}(i)$ the set of upstream neighbors of i (that is, $\mathcal{U}(i) = \{ j \mid (j,i)\in\mathcal{A} \}$). Consider the following problem

$$\text{Minimize} \quad \textstyle\sum_{(i,j)\in\mathcal{A}} a_{ij}(f_{ij}) \tag{4.1}$$

$$\text{Subject to} \quad \textstyle\sum_{j\in\mathcal{D}(i)} f_{ij} - \sum_{j\in\mathcal{U}(i)} f_{ji} = s_i, \ \forall\ i\in\mathcal{N}, \tag{4.2}$$

where each $a_{ij}:\Re\to\Re\cup\{+\infty\}$ is a strictly convex, lower semicontinuous function and each s_i is a real number (see Figure 4.1).

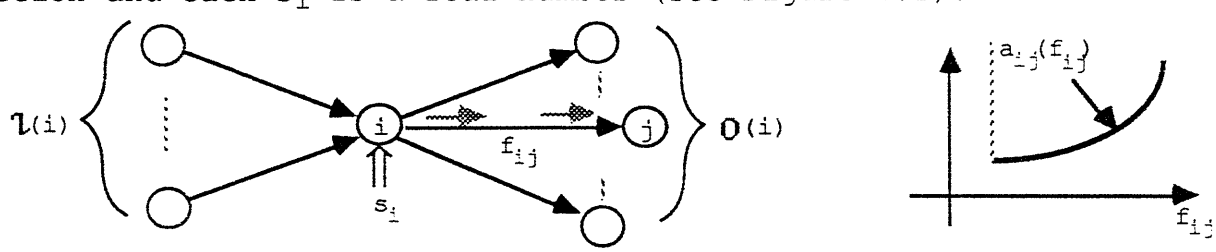


Figure 4.1

Note that any constraints of the form $Ef = b$, where each nonzero entry of E is either -1 or 1 and E has at most two nonzero entries per column, can be transformed into the conservation of flow constraint (4.2) by negating some columns and adding a dummy row containing 1 in columns where E has only one nonzero entry. Also note that capacity constraints of the form

$$b_{ij} \leq f_{ij} \leq c_{ij},$$

where b_{ij}, c_{ij} are given scalars, can be incorporated into the cost function a_{ij} by letting $a_{ij}(f_{ij}) = +\infty$ for $f_{ij}\notin [b_{ij},c_{ij}]$.

The problem of this section is an important optimization problem (see [2], [11], [22]-[25] for sequential algorithms and [1], [3], [26] for parallel algorithms).

Denote by $g_{ij}:\Re\to\Re\cup\{+\infty\}$ the <u>conjugate function</u> ([10], §12; [11], pp. 330) of a_{ij}, i.e.,

$$g_{ij}(\eta) = \sup_{\zeta\in\Re}\{\zeta\eta - a_{ij}(\zeta)\}.$$

Each g_{ij} is convex and, by assigning a Lagrange multiplier p_i (also called a <u>price</u>) to the ith constraint of (4.2), we can formulate the dual problem ([10], §8G) of (4.1) as the following convex minimization problem

$$\text{Minimize} \quad q(p) = \sum_{(i,j)\in\mathcal{A}} g_{ij}(p_i-p_j) - \sum_{i\in\mathcal{N}} p_i s_i \tag{4.3}$$

$$\text{Subject to } p\in\Re^n.$$

Let P^* be the set of optimal solutions for (4.3). We make the following assumption:

<u>Assumption E</u>

(a) Each g_{ij} is real valued.

(b) The set P^* is nonempty.

Assumption E implies (cf. [10], §11D) that the original problem (4.1) has an optimal solution and the optimal objective value for (4.1) and (4.3) sum to zero. Furthermore, the strict convexity of the a_{ij}'s implies that (4.1) has a <u>unique</u> optimal solution, which we denote by $f^* = (\ldots,f_{ij}^*,\ldots)_{(i,j)\in\mathcal{A}}$, and that every g_{ij} is continuously differentiable ([10], pp. 218, 253). Hence q given by (4.3) is also continuously differentiable and it can be seen from (4.3) that the partial derivative $\partial q(p)/\partial p_i$, to be denoted by $d_i(p)$, is given by

$$d_i(p) = \sum_{j\in\mathbf{D}(i)} \nabla g_{ij}(p_i-p_j) - \sum_{j\in\mathcal{U}(i)} \nabla g_{ji}(p_j-p_i) - s_i. \tag{4.5}$$

Given a price vector $p\in\Re^n$, we consider an iteration whereby the dual objective function q is minimized with respect to the ith coordinate p_i, while the remaining coordinates are held fixed. In view of the convexity and the differentiability of q, this is equivalent to solving the equation $d_i(p_1,\ldots,p_{i-1},\theta,p_{i+1},\ldots,p_n)=0$ with respect to the scalar θ. This equation can have several solutions and we will consider a mapping which chooses the solution which is nearest to the original price p_i. Accordingly, we define a function $h:\Re^n \rightarrow \Re^n$ whose ith coordinate is given by

$$h_i(p) = \operatorname{argmin}\{\, |\theta-p_i| \mid d_i(p_1,\ldots,p_{i-1},\theta,p_{i+1},\ldots,p_n)=0 \,\}. \tag{4.6}$$

The minimum in (4.6) is attained and h is well defined because the set $\{\, \theta \mid d_i(p_1,\ldots,p_{i-1},\theta,p_{i+1},\ldots,p_n)=0 \,\}$ is convex (due to the convexity of q) and closed (due to the continuity of d_i). Notice that $h(p) = p$ if and only if $\partial q(p)/\partial p_i = d_i(p) = 0$ for every i. It follows that P^* is the set of fixed points of h.

Since q is convex, the set P^* is convex (P^* is also nonempty by assumption). Also from Proposition 2.3 in [2] we have that, for any $p\in\Re^n$ and any $p^*\in P^*$,

$$\min_{j\in\mathcal{N}}\{p_j-p_j^*\} \le h_i(p)-p_i^* \le \max_{j\in\mathcal{N}}\{p_j-p_j^*\}, \ \forall\ i\in\mathcal{N},$$

and hence h has the pseudo-nonexpansive property

$$\|h(p)-p^*\| \le \|p-p^*\|.$$

Furthermore, by using Proposition 1 in [1] and an argument analogous to the proof of Proposition 2.5 in §7.2 of [3], we can show that the mapping h is continuous. Therefore h satisfies parts (a)-(c) of Assumption B. We show below that h also satisfies part (d) of Assumption B.

<u>Lemma 4.1</u> The mapping h satisfies Assumption B (d).

<u>Proof</u>: We start by mentioning certain facts that will be freely used in the course of the proof:

a) For any $(i,j)\in\mathcal{A}$, the function ∇g_{ij} is nondecreasing. (This is because g_{ij} is convex.)

b) $d_i:\Re^n \to \Re$ is a nondecreasing function of the ith coordinate of its argument when the other coordinates are held fixed. (This is because the dual functional q is convex and $d_i = \partial q/\partial p_i$.)

c) A vector $p^* \in \Re^n$ belongs to P^* if and only if, for every arc (i,j), we have $\nabla g_{ij}(p_i^*-p_j^*) = f_{ij}^*$. (This is a direct consequence of the Network Equilibrium Theorem in [11], pp. 349.)

Then for some $(p^*,\beta,S^-,S^+) \in \Omega$ there exists, for every $s \in (S^- \cup S^+)$, a $p^s \in F(p^*,\beta,S^-,S^+)$ such that $h_s(p^s) = p_s^s$ and $\rho(p^s,P^*) = \beta$. Let $S = S^- \cup S^+$ and $\varepsilon = \beta - \max\{\ |p_i^k-p_i^*|\ \mid i \notin S,\ k \in S\ \}$. Then, for all $s \in S$,

$$p_i^s \in [p_i^*-\beta+\varepsilon, p_i^*+\beta-\varepsilon] \text{ if } i \notin S, \tag{4.10}$$

$$p_i^s = p_i^*-\beta \text{ if } i \in S^-, \tag{4.11a}$$

$$p_i^s = p_i^*+\beta \text{ if } i \in S^+. \tag{4.11b}$$

Fix any $i \in S^-$. The relations (4.10), (4.11a) imply that

$$p_i^i-p_j^i \le (p_i^*-\beta)-(p_j^*-\beta) = p_i^*-p_j^*,\ \forall\, j \in \mathbf{D}(i),$$

$$p_j^i-p_i^i \ge (p_j^*-\beta)-(p_i^*-\beta) = p_j^*-p_i^*,\ \forall\, j \in \mathcal{U}(i),$$

and, since ∇g_{kl} is nonincreasing for all $(k,l) \in \mathcal{A}$,

$$\nabla g_{ij}(p_i^i-p_j^i) \le \nabla g_{ij}(p_i^*-p_j^*) = f_{ij}^*,\ \forall\, j \in \mathbf{D}(i), \tag{4.12a}$$

$$\nabla g_{ji}(p_j^i-p_i^i) \ge \nabla g_{ji}(p_j^*-p_i^*) = f_{ji}^*,\ \forall\, j \in \mathcal{U}(i). \tag{4.12b}$$

Since $i \in S^-$, we have $h_i(p^i) = p_i^i$ or, equivalently, $d_i(p^i) = 0$. Then (4.5) and (4.12a)-(4.12b) imply that

$$\begin{aligned} 0 &= d_i(p^i) \\ &= \sum_{j \in \mathbf{D}(i)} \nabla g_{ji}(p_i^i-p_j^i) - \sum_{j \in \mathcal{U}(i)} \nabla g_{ji}(p_j^i-p_i^i) + s_i \\ &\ge \sum_{j \in \mathbf{D}(i)} f_{ij}^* - \sum_{j \in \mathcal{U}(i)} f_{ji}^* + s_i \\ &= 0, \end{aligned}$$

where the last equality follows because the flows f_{ij}^* and f_{ji}^* must satisfy the flow conservation equation (4.2). It follows that the inequalities in (4.12a)-(4.12b) are actually equalities and

$$\nabla g_{ij}(p_i{}^i-p_j{}^i) = f_{ij}{}^*, \ \forall\, j\in\mathcal{D}(i), \tag{4.13a}$$

$$\nabla g_{ji}(p_j{}^i-p_i{}^i) = f_{ji}{}^*, \ \forall\, j\in\mathcal{U}(i). \tag{4.13b}$$

Since the choice of $i\in S^-$ was arbitrary, (4.13a)-(4.13b) hold for all $i\in S^-$. By an analogous argument (using (4.11b) instead of (4.11a)) we can show that (4.13a)-(4.13b) hold for all $i\in S^+$ as well.

Let $\pi\in\Re^n$ be the vector whose ith component is

$$\pi_i = \begin{cases} p_i{}^* + \varepsilon & \text{if } i\in S^+, \\ p_i{}^* - \varepsilon & \text{if } i\in S^-, \\ p_i{}^* & \text{otherwise.} \end{cases} \tag{4.14}$$

We claim that

$$\nabla g_{ij}(\pi_i-\pi_j) = f_{ij}{}^*, \ \forall\,(i,j)\in\mathcal{A}. \tag{4.15}$$

To see this, we note from (4.10), (4.11a)-(4.11b), (4.14) and the fact $\varepsilon\le\beta$ that, for any $(i,j)\in\mathcal{A}$,

$$\begin{aligned}
&\pi_i-\pi_j = p_i{}^*-p_j{}^*, && \text{if } i\notin S,\ j\notin S,\\
&\pi_i-\pi_j = (p_i{}^*+\varepsilon)-(p_j{}^*+\varepsilon) = p_i{}^*-p_j{}^*, && \text{if } i\in S^+,\ j\in S^+,\\
&\pi_i-\pi_j = (p_i{}^*-\varepsilon)-(p_j{}^*-\varepsilon) = p_i{}^*-p_j{}^*, && \text{if } i\in S^-,\ j\in S^-,\\
&p_i{}^i-p_j{}^i = (p_i{}^*+\beta)-(p_j{}^*-\beta) \ge \pi_i-\pi_j \ge p_i{}^*-p_j{}^*, && \text{if } i\in S^+,\ j\in S^-,\\
&p_i{}^i-p_j{}^i = (p_i{}^*-\beta)-(p_j{}^*+\beta) \le \pi_i-\pi_j \le p_i{}^*-p_j{}^*, && \text{if } i\in S^-,\ j\in S^+,\\
&p_i{}^i-p_j{}^i \ge (p_i{}^*+\beta)-(p_j{}^*+\beta-\varepsilon) = \pi_i-\pi_j \ge p_i{}^*-p_j{}^*, && \text{if } i\in S^+,\ j\notin S,\\
&p_i{}^i-p_j{}^i \le (p_i{}^*-\beta)-(p_j{}^*-\beta+\varepsilon) = \pi_i-\pi_j \le p_i{}^*-p_j{}^*, && \text{if } i\in S^-,\ j\notin S,\\
&p_i{}^i-p_j{}^i \le (p_i{}^*+\beta-\varepsilon)-(p_j{}^*+\beta) = \pi_i-\pi_j \le p_i{}^*-p_j{}^*, && \text{if } i\notin S,\ j\in S^+,\\
&p_i{}^i-p_j{}^i \ge (p_i{}^*-\beta+\varepsilon)-(p_j{}^*-\beta) = \pi_i-\pi_j \ge p_i{}^*-p_j{}^*, && \text{if } i\notin S,\ j\in S^-.
\end{aligned}$$

Consider any $(i,j)\in\mathcal{A}$. The preceding inequalities show that $\pi_i-\pi_j$ is always between $p_i{}^i-p_j{}^i$ and $p_i{}^*-p_j{}^*$. The monotonicity of ∇g_{ij} and the equalities $\nabla g_{ij}(p_i{}^*-p_j{}^*) = f_{ij}{}^* = \nabla g_{ij}(p_i{}^i-p_j{}^i)$ (cf. Eq. (4.13)) imply that $\nabla g_{ij}(\pi_i-\pi_j) = f_{ij}{}^*$. This completes the proof of (4.15).

Eq. (4.15) implies that $\pi \in P^*$. Since (cf. (4.10), (4.11), (4.14)) $\|p^s-\pi\| < \|p^s-p^*\|$ for all $s \in S$, this contradicts the hypothesis that $\rho(p^s,P^*) = \|p^s-p^*\|$ for all $s \in S$. Q.E.D.

Since h has been shown to satisfy Assumption B, we conclude from Lemma 2.2 and Proposition 2.1 that the partially asynchronous iteration

$$p := (1-\gamma)p + \gamma h(p)$$

converges to an optimal price vector p^*, where $\gamma \in (0,1)$. The optimal flows are obtained as a byproduct, using the relation $\nabla g_{ij}(p_i{}^*-p_j{}^*) = f_{ij}{}^*$. Notice that the iteration for each coordinate i consists of minimization along the ith coordinate direction (to obtain $h_i(p)$) followed by the use of the relaxation parameter γ to obtain the new value $(1-\gamma)p_i + \gamma h_i(p)$. As a special case, we have that the synchronous Jacobi algorithm

$$p(t+1) = (1-\gamma)p(t) + \gamma h(p(t))$$

is also convergent, which is a new result.

A related result can be found in [1] where totally asynchronous convergence is established even if $\gamma = 1$, provided that a particular coordinate of p is never iterated upon. An experimental comparison of the two methods will be presented in §8. We remark that the results in this section also extend to the case where each arc has a gain of either +1 or -1 (i.e., the f_{ji} term in Eq. (4.2) is multiplied by either a +1 or a -1).

5. Agreement and Markov Chain Algorithms

In this section we consider two problems: a problem of agreement and the computation of the invariant distribution of a Markov chain. These problems are the only ones for which partially asynchronous algorithms that converge for every value of the asynchronism bound B of Assumption A are available [13], [15] (in fact, these algorithms have been shown to converge at a geometric rate). We show that these results can also be obtained by applying our general convergence theorem (Proposition 2.1).

5.1. The Agreement Algorithm

We consider here a set of n processors, numbered from 1 to n, who try to reach agreement on a common value by exchanging tentative values and forming convex combinations of their own values with the values received from other processors. This algorithm has been used in [14]-[15] in the context of asynchronous stochastic gradient methods with the purpose of averaging noisy measurements of the same variable by different processors.

We now formally describe the agreement algorithm. Let $N = \{1,\ldots,n\}$. Each processor i has a set of nonnegative coefficients $\{a_{i1},\ldots,a_{in}\}$ satisfying $a_{ii} > 0$, $\sum_{j\in N} a_{ij} = 1$, and at time t it possesses estimate $x_i(t)$ which is updated according to (cf. (1.1))

$$x_i(t+1) = \begin{cases} \sum_{j\in N} a_{ij}\cdot x_j(\tau_{ij}(t)) & \text{if } t\in T_i, \\ x_i(t) & \text{otherwise.} \end{cases} \tag{5.1a}$$

$$x_i(1-B) = \ldots = x_i(0) = \overline{x}_i, \tag{5.1b}$$

where T_i and $\tau_{ij}(t)$ are as in §1 and $\overline{x}_i$ is the initial value of processor i. Let A be the n×n matrix whose (i,j)th entry is a_{ij} and let γ be a positive lower bound on the a_{ii}'s. Using the results from §1 to §3 we obtain the following:

<u>Proposition 5.1</u> If A is irreducible and Assumption A holds, then $\{x_i(t)\} \to y$ for all $i \in N$, where y is some scalar between $\min_i\{\overline{x}_i\}$ and $\max_i\{\overline{x}_i\}$.

<u>Proof</u>: It can be seen that (5.1a) is a special case of (1.1) with $f(x) = Ax$. Let

$$B = (A-\gamma I)/(1-\gamma).$$

Then $\gamma \in (0,1)$,

$$A = \gamma I + (1-\gamma)B, \tag{5.2}$$

and $B = [b_{ij}]$ can be seen to satisfy $\sum_{j \in N} |b_{ij}| \le 1$. Moreover, since A is irreducible, so is B. Hence the function $h:\Re^n \to \Re^n$ defined by $h(x) = Bx$ satisfies Assumption D in §3. Since h has a fixed point (the zero vector), this together with Proposition 3.1 implies that h satisfies Assumption B. Since (cf. (5.2)) $f(x) = \gamma x + (1-\gamma)h(x)$, this together with Lemma 2.2 shows that f satisfies Assumption C. Then by Proposition 2.1 the sequence $\{x(t)\}$ generated by (5.1a)-(5.1b) converges to some point x^∞ satisfying $Ax^\infty = x^\infty$. Since A is stochastic, x^∞ must be of the form $(y,\ldots,y)$ for some $y \in \Re$. It can be seen from (5.1b) that, for $r \in \{1-B,\ldots,0\}$,

$$x_i(r) \le \max_j\{\overline{x}_j\}, \quad \forall\ i \in N. \tag{5.3}$$

Suppose that (5.3) holds for all $r \in \{1-B,\ldots,t\}$, for some $t \ge 0$. Then by (5.1a) and the property of the a_{ij}'s,

$$\begin{aligned} x_i(t+1) &= \sum_{j \in N} a_{ij} \cdot x_j(\tau_{ij}(t)) \\ &\le \sum_{j \in N} a_{ij} \cdot \max_j\{\overline{x}_j\} \\ &= \max_j\{\overline{x}_j\}, \end{aligned}$$

for all i such that $t \in \mathbf{T}_i$, and $x_i(t+1) = x_i(t) \le \max_j\{\overline{x}_j\}$ for all other i. Hence, by induction, (5.3) holds for all $r \in \{1-B, 2-B,\ldots\}$. Since $x_i(r) \to y$ for each i, this implies that $y \le \max_j\{\overline{x}_j\}$. A symmetrical argument shows $y \geq \min_i\{\bar{x}_i\}$.   Q.E.D.

It can be shown [3], [15] that Proposition 5.1 remains valid if a_{ii} is positive for at least one (but not all) i and, furthermore, convergence takes place at the rate of a geometric progression. The proof however is more complex. Similar results can be found in [15] for more general versions of the agreement algorithm.

5.2. Invariant Distribution of Markov Chains

Let P be an irreducible stochastic matrix of dimension n×n. We denote by p_{ij} the (i,j)th entry of P and we assume that $p_{ii} > 0$ for all i. We wish to compute a row vector π^* of invariant probabilities for the corresponding Markov chain, i.e., $\pi_i^* \geq 0$, $\sum_{i\in N} \pi_i^* = 1$, $\pi^* = \pi^* P$, where $N = \{1,\ldots,n\}$. (We actually have $\pi_i^* > 0$, for all i, due to the irreducibility of P [31].) As in §5.1, suppose that we have a network of n processors and that the ith processor ($i\in N$) generates a sequence of estimates $\{\pi_i(t)\}$ using the following partially asynchronous version of the classical serial algorithm $\pi := \pi P$ (cf. (5.1a)-(5.1b)):

$$\pi_i(t+1) = \begin{cases} \sum_{j\in N} p_{ji}\pi_i(\tau_{ij}(t)) & \text{if } t\in T_i, \\ \pi_i(t) & \text{otherwise.} \end{cases} \tag{5.4}$$

$$\pi_i(1-B) = \ldots = \pi_i(0),$$

where T_i and $\tau_{ij}(t)$ are as in §1 and $\pi_i(0)$ is any positive scalar. This asynchronous algorithm was introduced in [12], where geometric convergence was established. We show below that convergence also follows from our general results.

Proposition 5.3 If Assumption A holds, then then there exists a positive number c such that $\pi(t) \to c \cdot \pi^*$.

Proof: We will show that (5.4) is a special case of (5.1a). Let

$$x_i(t) = \pi_i(t)/\pi_i^*, \qquad a_{ij} = \pi_j^* \cdot p_{ji}/\pi_i^*. \tag{5.5}$$

Then the matrix $A = [a_{ij}]$ is nonnegative and irreducible, has positive diagonal entries, and

$$\begin{aligned} \sum_{j \in N} a_{ij} &= \sum_{j \in N} \pi_j^* \cdot p_{ji}/\pi_i^* \\ &= \pi_i^*/\pi_i^* \\ &= 1, \end{aligned}$$

where the second equality follows from $\pi^* = \pi^* P$. Furthermore it can be seen from (5.4) and (5.5) that $x_i(t)$ evolves according to the iteration (5.1). Therefore, by Proposition 5.1 and the initial positivity conditions, $\{x_i(t)\} \to c$ for all i, where c is some positive scalar. It follows from (5.5) that $\pi_i(t) \to c \cdot \pi_i^*$ for all i. Q.E.D.

Upon obtaining $c \cdot \pi^*$, π^* can be recovered by normalizing $c \cdot \pi^*$.

6. Neural Networks

Consider a connected, directed network $\mathcal{G} = (\mathcal{N},\mathcal{A})$ and, for each $i\in\mathcal{N}$, denote by $\mathcal{U}(i)$ the set $\{ j \mid (j,i)\in\mathcal{A} \}$ of upstream neighbors of i. Let $\{\sigma_i\}_{i\in\mathcal{N}}$ be a set of given scalars and let $\{\lambda_{ij}\}_{j\in\mathcal{U}(i)}$ be a set of nonzero scalars satisfying $\sum_{j\in\mathcal{U}(i)} |\lambda_{ij}| \le 1$. We wish to find scalars $\{x_i\}_{i\in\mathcal{N}}$ such that

$$x_i = \phi_i\left(\sum_{j\in\mathcal{U}(i)} \lambda_{ij}x_j + \sigma_i\right), \quad \forall\ i\in\mathcal{N}, \tag{6.1}$$

where $\phi_i:\Re\to\Re$ is a continuous nondecreasing function satisfying

$$\lim_{\xi\to-\infty} \phi_i(\xi) = -1, \quad \lim_{\xi\to+\infty} \phi_i(\xi) = 1, \tag{6.2}$$

(see Figure 6.1). Notice that the function ϕ_i maps the set $[-1,1]^n$ into itself and, by Brouwer's fixed point theorem ([6], pp. 17), the system (6.1) is guaranteed to have a solution.

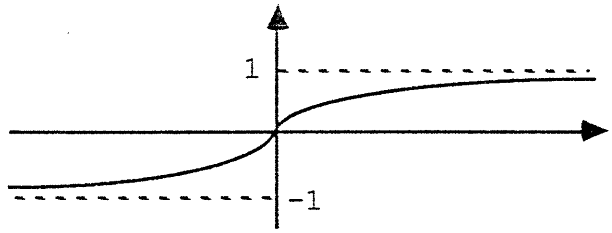


Figure 6.1. The function ϕ_i.

If we think of each node i as a neuron, Eqs. (6.1) and (6.2) imply that this neuron is turned on (i.e., $x_i \approx 1$) if the majority of its inputs are also turned on. Thus x_i gives the state ("on" or "off") of the ith neuron for a given set of connections (specified by $\mathcal{A}$) and a given external excitation (specified by σ_i) (see Figure 6.2).

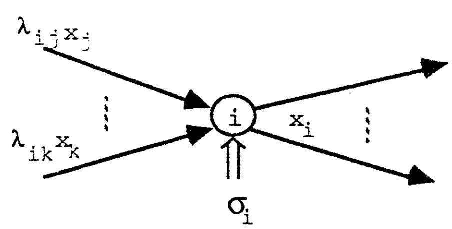


Figure 6.2.

Indeed, (6.1) and (6.2) describe a class of neural networks that have been applied to solving a number of problems in combinatorial optimization, pattern recognition and artificial intelligence [28]-[30].

Let $f:\Re^n \to \Re^n$ be the function whose ith component is

$$f_i(x) = \phi_i\left(\sum_{j\in\mathcal{U}(i)} \lambda_{ij}x_j + \sigma_i\right), \quad \forall\ i\in\mathcal{N}. \tag{6.3}$$

Then solving (6.1) is equivalent to finding a fixed point of f. In what follows, we consider a special form for ϕ_i and show that it gives rise, in a natural way, to a nonexpansive function f that satisfies Assumptions B and C of §2. To the best of our knowledge, asynchronous convergence of neural networks has not been explored before. In some sense, asynchronous neural networks are quite natural since biological neural connections may experience long propagation delay.

Let ϕ_i^+ denote the right derivative of ϕ_i, i.e.,

$$\phi_i^+(\xi) = \lim_{\varepsilon\downarrow 0}(\phi_i(\xi+\varepsilon)-\phi_i(\xi))/\varepsilon, \quad \forall\ \xi\in\Re.$$

The following result shows that, if ϕ_i^+ is sufficiently small for all i, then f given by (6.3) satisfies Assumption B′.

Proposition 6.1 If $\mathcal{G}$ is strongly connected and each ϕ_i is continuous, satisfies (6.2) and

$$0 \le \phi_i^+(\xi) \le 1, \quad \forall\ \xi\in\Re, \tag{6.4}$$

then f given by (6.3) satisfies Assumption B′.

Proof: We have seen earlier that f has a fixed point. Since each ϕ_i is continuous, f is also continuous. Now we will show that f is nonexpansive. Fix any $i\in\mathcal{N}$. Since (cf. (6.4)) the slope of ϕ_i is bounded inside the interval [0,1], we have

$$|\phi_i(b)-\phi_i(a)| \le |b-a|, \quad \forall\ a\in\Re,\ b\in\Re.$$

Hence, for any $x\in\Re^n$ and $y\in\Re^n$,

$$
\begin{aligned}
|f_i(y)-f_i(x)| &= |\phi_i(\textstyle\sum_{j\in\mathcal{U}(i)} \lambda_{ij}y_j+\sigma_i)-\phi_i(\sum_{j\in\mathcal{U}(i)} \lambda_{ij}x_j+\sigma_i)| \\
&\le |\textstyle\sum_{j\in\mathcal{U}(i)} \lambda_{ij}(y_j-x_j)| \\
&\le \textstyle\sum_{j\in\mathcal{U}(i)} |\lambda_{ij}||y_j-x_j|. \qquad (6.5)
\end{aligned}
$$

Since $\sum_{j\in\mathcal{U}(i)}|\lambda_{ij}| \le 1$, Eq. (6.5) implies that

$$|f_i(y)-f_i(x)| \le \|x-y\|.$$

Since the choice of $i\in\mathcal{N}$ was arbitrary, this in turn implies that

$$\|f(x)-f(y)\| \le \|x-y\|, \ \forall\ x\in\Re^n,\ y\in\Re^n.$$

Therefore f is nonexpansive.

It remains to show that f satisfies Assumption B′ (d). Suppose the contrary. Then there exists an $(x^*,\beta,S^-,S^+)\in\Omega$ such that, for every $s\in(S^-\cup S^+)$, there is an $x^s\in F(x^*,\beta,S^-,S^+)$ such that

$$x^s\notin X^* \quad\text{and}\quad f_s(x^s) = x_s{}^s.$$

Let $S = S^-\cup S^+$ ($S \ne \mathcal{N}$ since $x^s\notin X^*$ for all $s\in S$) and fix any $i\in S$. By (6.5) and the fact $x^* = f(x^*)$, we obtain that

$$|f_i(x^i)-f_i(x^*)| \le \textstyle\sum_{j\in\mathcal{U}(i)}|\lambda_{ij}||x_j{}^i-x_j{}^*|.$$

Hence

$$
\begin{aligned}
\beta &\le \textstyle\sum_{j\in\mathcal{U}(i)}|\lambda_{ij}||x_j{}^i-x_j{}^*| \\
&= \textstyle\sum_{j\in\mathcal{U}(i)}|\lambda_{ij}|\beta+\sum_{j\in\mathcal{U}(i),\, j\notin S}|\lambda_{ij}|(|x_j{}^i-x_j{}^*|-\beta) \\
&\le \beta+\textstyle\sum_{j\in\mathcal{U}(i),\, j\notin S}|\lambda_{ij}|(|x_j{}^i-x_j{}^*|-\beta).
\end{aligned}
$$

Since $|x_j{}^i-x_j{}^*|<\beta$ and $\lambda_{ij} \ne 0$ for all $j\in\mathcal{U}(i)$, $j\notin S$, this implies that $\mathcal{U}(i)\cap(\mathcal{N}\backslash S) = \emptyset$. Since the choice of $i\in S$ was arbitrary, it follows that $\mathcal{U}(i)\cap(\mathcal{N}\backslash S) = \emptyset$ for all $i\in S$. Hence $\mathcal{G}$ is not strongly connected, a contradiction of our hypothesis. Q.E.D.

It follows from Lemma 2.2 and Propositions 6.1, 2.1 that the asynchronous neural iteraton

$$x_i := (1-\gamma)x_i + \gamma \cdot \phi_i\left(\sum_{j\in\mathcal{U}(i)} \lambda_{ij}x_j + \sigma_i\right)$$

converges, where $\gamma\in(0,1)$ is a relaxation parameter. Two examples of ϕ_i that satisfy the hypothesis of Proposition 6.1 are

$$\phi_i(\xi) = 2(1+e^{-2\xi})^{-1} - 1,$$

and

$$\phi_i(\xi) = \max\{-1,\min\{1,\xi\}\}.$$

Let us briefly discuss an alternative form for the function ϕ_i. If we assume that each ϕ_i is continuously differentiable and its derivative ϕ_i' satisfies $0 < \phi_i'(\xi) < 1$ for all $\xi\in\Re$, then it can be shown that the restriction of the function f on a compact set is a contraction. In that case, the asynchronous neural iteration

$$x_i := \phi_i\left(\sum_{j\in\mathcal{U}(i)} \lambda_{ij}x_j + \sigma_i\right)$$

can be shown to converge even under the total asynchronism assumption

$$\lim_{t\to+\infty} \tau_{ij}(t) = +\infty, \quad \forall\ i,\ j$$

(cf. Proposition 2.1 in §6.2 of [3]).

7. Least Element of Weakly Diagonally Dominant, Leontief Systems

Let $A = [a_{kj}]$ be a given $m\times n$ matrix (with $m \geq n$) and $b = (\ldots,b_k,\ldots)$ be an element of $\Re^m$. We make the following assumption:

Assumption F

(a) Each row of A has exactly one positive entry and the set

$$I(i) \equiv \{\ k \mid a_{ki} > 0\ \}$$

is nonempty for all $i\in N$ (i.e., every column has at least one positive entry).

(b) $-\sum_{j\neq i} a_{kj} \le a_{ki}$ if $a_{ki} > 0$.

(c) For any $(k_1,\ldots,k_n)\in I(1)\times\ldots\times I(n)$, the matrix $[a_{k_ij}]_{i\in N, j\in N}$ is irreducible.

Since $a_{ki} > 0$ for all $k\in I(i)$, we will, by dividing the kth constraint by a_{ki} if necessary, assume that $a_{ki} = 1$ for all $k\in I(i)$, in which case parts (a) and (b) of Assumption F are equivalent to

$$a_{ki} = 1, \quad -\sum_{j\neq i} a_{kj} \le 1 \quad \text{and} \quad a_{kj} \le 0, \; \forall\; j\neq i, \tag{7.1}$$

for all $k\in I(i)$, $i\in N$.

Let X be the polyhedral set

$$X = \{\; x\in\Re^n \mid Ax \ge b \;\}. \tag{7.2}$$

We wish to find an element η of X satisfying

$$x \ge \eta, \; \forall\; x\in X$$

(such an element is called the least element of X in [7] and [8]). Notice that if a least element exists, then it is unique. Let $h:\Re^n\to\Re^n$ be the function whose ith component is

$$h_i(x) = \max_{k\in I(i)}\{\; b_k - \sum_{j\neq i} a_{kj}x_j \;\}. \tag{7.3}$$

It is shown in [7] that X has a least element for all b such that X is nonempty if and only if A^T is Leontief (a matrix E is Leontief if each column of E has at most one positive entry and there exists $y \ge 0$ such that $Ey > 0$ componentwise). The following lemma sharpens this result by giving a simpler necessary and sufficient condition for X to have a least element. It also relates the least element of X to the fixed points of h.

Lemma 7.1 Suppose that $X \neq \varnothing$ and that Assumption F holds. Then,

(a) X has no least element if and only if

$$\sum_{j\in N} a_{kj} = 0, \; \forall\, k. \tag{7.4}$$

(b) If η is a least element of X, then it is a fixed point of h.

Proof: We first prove (a). Suppose that (7.4) holds and let $e \in \Re^n$ be the vector with all coordinates equal to 1. Eq. (7.4) shows that $Ae = 0$. Thus, if x is an element of X, then $x-\lambda e \in X$, for all positive scalars λ. It follows that X cannot have a least element. Now suppose that (7.4) does not hold. We first show that X is bounded from below (i.e., there exists some $a \in \Re^n$ such that $x \geq a$ componentwise for all $x \in X$). If this were not so, then there would exist some $v \in \Re^n$ and some $x \in X$ such that $v_i < 0$ for some i and $x+\lambda v \in X$ for all positive scalars λ. The latter implies that $A(x+\lambda v) \geq b$ for all $\lambda > 0$ and hence $Av \geq 0$. Fix any scalars $(k_1,\ldots,k_n) \in I(1)\times\ldots\times I(n)$ and consider an i such that $v_i = \min_j\{v_j\}$. Then (cf. $Av \geq 0$ and (7.1))

$$0 \leq v_i + \sum_{j\neq i} a_{k_ij}v_j = (1-\sum_{j\neq i} |a_{k_ij}|)v_i + \sum_{j\neq i} |a_{k_ij}|(v_i-v_j).$$

Since $v_i < 0$ and $v_i-v_j \leq 0$ for all $j \neq i$, this implies that $\sum_{j\neq i} |a_{k_ij}| = 1$ and $v_i = v_j$ for all $j \neq i$ such that $a_{k_ij} \neq 0$. By Assumption F (c), there exists $j \neq i$ such that $a_{k_ij} \neq 0$. We then repeat the above argument with j in place of i. In this way, we eventually obtain that $v_1 = \ldots = v_n$ and $1 = \sum_{j\neq i} |a_{k_ij}|$ for all $i \in N$. Since our choice of $(k_1,\ldots,k_n) \in I(1)\times\ldots\times I(n)$ was arbitrary, (7.4) holds - contradicting our hypothesis. Hence X is bounded from below. Using (7.1), it is easily verified that if x' and x'' are two elements of X, then the n-vector x whose ith component is $\min\{x_i', x_i''\}$ is also an element of X. Since X is closed and bounded from below, X has a least element.

We next prove (b). Since $\eta \in X$, we have

$$\sum_{j\neq i} a_{kj}\eta_j + \eta_i \geq b_k,\ \forall\ k \in I(i),\ \forall\ i \in N.$$

Thus,

$$h_i(\eta) = \max_{k\in I(i)}\{b_k - \sum_{j\neq i} a_{kj}\eta_j\} \leq \eta_i, \ \forall\ i\in N.$$

If η is not a fixed point of h, then the set $I = \{\ i\in N \mid h_i(\eta) < \eta_i\ \}$ is nonempty. Then, for every $i\in I$, we have

$$\sum_{j\in N} a_{kj}\eta_j > b_k, \ \forall\ k\in I(i). \tag{7.5}$$

Consider the n-vector $\tilde{\eta}$, defined by $\tilde{\eta}_i = \eta_i-\varepsilon$, if $i\in I$, and $\tilde{\eta}_i = \eta_i$, otherwise. For ε positive and small enough, the inequalities (7.5) remain valid. Furthermore if $k\notin\cup_{i\in I}I(i)$, we have

$$\sum_{i\in N} a_{ki}\tilde{\eta}_i = \sum_{i\notin I} a_{ki}\eta_i + \sum_{i\in I} a_{ki}(\eta_i-\varepsilon) \geq \sum_{i\in N} a_{ki}\eta_i \geq b_k,$$

where we used the property $a_{ki} \leq 0$ for $k\notin I(i)$. Thus, $\tilde{\eta}\in X$, contradicting the hypothesis that η is the least element of X.
Q.E.D.

Let X^* denote the set of fixed points of h. Suppose that X^* is nonempty (Lemma 7.1 gives sufficient conditions for X^* to be nonempty). We will show that h satisfies Assumption B′. Since (cf. (7.1)) h is continuous, it suffices to show that parts (c) and (d) of Assumption B′ hold.

<u>Lemma 7.2</u> $\|h(x)-h(y)\| \leq \|x-y\|$ for any $x\in\Re^n$ and any $y\in\Re^n$.

<u>Proof</u>: Let $z = h(x)$, $w = h(y)$ and consider any $i\in N$. We will show that $|z_i-w_i| \leq \|x-y\|$, from which our claim follows. Since $z_i = h_i(x)$ and $w_i = h_i(y)$, it follows from (7.3) that, for some k in I(i),

$$\sum_{j\neq i} a_{kj}x_j + z_i \geq b_k, \tag{7.5a}$$

$$\sum_{j\neq i} a_{kj}y_j + w_i = b_k, \tag{7.5b}$$

Subtracting (7.5b) from (7.5a), we obtain

$$\sum_{j\neq i} a_{kj}(x_j-y_j) + (z_i-w_i) \geq 0.$$

This together with (7.1) implies that

$$\begin{aligned} w_i - z_i &\le \sum_{j\ne i} a_{kj}(x_j - y_j) \\ &\le \sum_{j\ne i} |a_{kj}| \cdot \|x-y\| \\ &\le \|x-y\|. \end{aligned}$$

The inequality $z_i - w_i \le \|x-y\|$ is obtained similarly. Q.E.D.

Lemma 7.3 h satisfies Assumption B′ (d).

Proof: Suppose the contrary. Then for some $(x^*,\beta,S^-,S^+)\in\Omega$ there exists, for every $s\in(S^-\cup S^+)$, an $x^s\in F(x^*,\beta,S^-,S^+)$ such that

$$x^s \notin X^* \quad\text{and}\quad h_s(x^s) = x_s{}^s.$$

Let $S = S^-\cup S^+$. We must have $S \ne N$ because otherwise the set $F(x^*,\beta,S^-,S^+)$ would be a singleton, implying that all the vectors x^s, $s\in N$, are equal, in which case each x^s is a fixed point of h, a contradiction.

Fix any $i\in S^-$. By (7.3) and the hypothesis $x^* = h(x^*)$, there exists some $k_i\in I(i)$ such that

$$\sum_{j\in N} a_{k_i j} x_j{}^* = b_{k_i}. \tag{7.6}$$

Since $x_i = h_i(x^i)$, $\sum_{j\in N} a_{k_i j} x_j{}^i \ge b_{k_i}$. It then follows from (7.6) that

$$\sum_{j\in N} a_{k_i j}(x_j{}^i - x_j{}^*) \ge 0.$$

This implies (using (7.1) and the facts $k_i\in I(i)$, $i\in S^-$) that

$$\begin{aligned} 0 &\le -\beta\sum_{j\in S^-} a_{k_i j} + \beta\sum_{j\in S^+} a_{k_i j} + \sum_{j\notin S} |a_{k_i j}|\,|x_j{}^i - x_j{}^*| \\ &= -\beta\sum_{j\in S^-} a_{k_i j} - \beta\sum_{j\in S^+} |a_{k_i j}| + \sum_{j\notin S} |a_{k_i j}|\,|x_j{}^i - x_j{}^*| \\ &= -\beta(1 - \sum_{j\ne i} |a_{k_i j}|) - 2\beta\sum_{j\in S^+} |a_{k_i j}| + \sum_{j\notin S} |a_{k_i j}|(|x_j{}^i - x_j{}^*| - \beta). \end{aligned}$$

Since $|x_j{}^i - y_j| < \beta$ for all $j\notin S$, (7.1) implies that

$$\sum_{j\neq i} a_{k_i j} = -1 \quad \text{and} \quad a_{k_i j} = 0, \ \forall\, j\notin S^-. \tag{7.7}$$

Since the choice of i was arbitrary, (7.7) holds for all $i\in S^-$. By an analogous argument, we obtain that, for all $i\in S^+$,

$$\sum_{j\neq i} a_{k_i j} = -1 \quad \text{and} \quad a_{k_i j} = 0, \ \forall\, j\notin S^+, \tag{7.8}$$

where each k_i is a scalar in $I(i)$ such that

$$\sum_{j\in N} a_{k_i j}\, x_j{}^i = b_{k_i}.$$

For each $i\notin S$, let k_i be any element of $I(i)$. Since $S\neq N$, (7.7) and (7.8) imply that the matrix $[a_{k_i j}]_{i\in N, j\in N}$ is <u>not</u> irreducible - a contradiction of Assumption F (c). Q.E.D.

We may now invoke Lemma 2.2 and Proposition 2.1 to establish that the partially asynchronous iteration $x := (1-\gamma)x+\gamma h(x)$ converges to a fixed point of h. Unfortunately, such a fixed point is not necessarily the least element of X. We have, however, the following characterization of such fixed points:

<u>Lemma 7.4</u> If X has a least element η, then, for any fixed point x^* of h, there exists a nonnegative scalar λ such that $x^* = \eta+(\lambda,\ldots,\lambda)$.

<u>Proof</u>: Since x^* is a fixed point of h, $x^*\in X$. Hence $x^* \geq \eta$. We then repeat the proof of Lemma 7.3, with $S^- = N$ and $x^i = \eta$ for all $i\in N$. This yields $x_i{}^*-\eta_i \leq \sum_{j\neq i} |a_{k_i j}|(x_j{}^*-\eta_j)$ for all $i\in N$. Since $x^*-\eta \geq 0$, Assumption F implies that the $x_i{}^*-\eta_i$'s are equal. Q.E.D.

Lemma 7.4 states that, given a fixed point x^* of h, we can compute the least element of X by a simple line search along the direction $(-1,\ldots,-1)$ (the stepsize λ is the largest for which $x^*-(\lambda,\ldots,\lambda)$ is in X). An example of X for which the corresponding h has multiple fixed

points is

$$X = \{ (x_1, x_2) \mid x_1 - x_2 \ge 0,\ x_1 - x_2/2 \ge -1,\ -x_1 + x_2 \ge 0 \}.$$

Here $h_1(x) = \max\{x_2, -1+x_2/2\}$, $h_2(x) = x_1$ and both $(-1,-1)$ and $(-2,-2)$ are fixed points of h (the least element of X is $(-2,-2)$).

Let us remark that if the inequalities in Assumption F are strict, then the mapping h is a contraction mapping (the same argument as in Lemma 7.2) and totally asynchronous convergence is obtained. We also remark that, if in the statement of Assumption F (c) we replace "For any" by the weaker "For some", then Lemmas 7.1 and 7.2 still hold, but not Lemmas 7.3 and 7.4. In fact, it can be shown that X^* is not necessarily convex in this case.

8. Simulation for Network Flow Problems

In this section we study and compare, using simulation, the performance of synchronous and partially asynchronous algorithms for the network flow problem of §4. We measure the following: (a) the effects of the stepsize γ (cf. Lemma 2.2), the problem size n, and the asynchrony measure B on the performance of partially asynchronous algorithms, (b) the efficiency of different partially asynchronous algorithms relative to each other and also relative to the corresponding synchronous algorithms.

In our study, we consider a special case of the network flow problem (4.1)-(4.2) where each cost function $a_{ij}(\cdot)$ is a quadratic on $[0,+\infty]$, i.e.,

$$a_{ij}(f_{ij}) = \begin{cases} \alpha_{ij}|f_{ij}|^2 + \beta_{ij}f_{ij} & \text{if } f_{ij} \ge 0, \\ +\infty & \text{otherwise,} \end{cases} \tag{8.1}$$

where α_{ij} is a given positive scalar and β_{ij} is a given scalar. This special case has many practical applications and has been studied extensively [22]-[25]. In what follows, we will denote by $h:\Re^n \rightarrow \Re^n$ the function given by (4.6) and (8.1). All of the algorithms involved in our study are based on h.

8.1. Test Problem Generation

In our test, each α_{ij} is randomly generated from the interval [1,5] and each β_{ij} is randomly generated from the set {1,2,...,100}. The average node degree is 10, i.e., $|\mathcal{A}| = 10\cdot n$, and the average node supply is 1000, i.e., $\sum_{i\in\mathcal{N}}|s_i| = 1000\cdot n$. Half of the nodes are supply nodes and half of the nodes are demand nodes (a node i is a supply (demand) node if $s_i > 0$ ($s_i < 0$)). The problems are generated using the linear cost network generator NETGEN [21], modified to generate quadratic cost coefficients as well.

8.2. The Main Partially Asynchronous Algorithm

The main focus of our study is the partially asynchronous algorithm described in §4. This algorithm, called PASYN, generates a sequence {x(t)} using the partially asynchronous iteration (1.1) under Assumption A, where the algorithmic mapping f is given by

$$f(x) = (1-\gamma)x+\gamma h(x). \tag{8.2}$$

In our simulation, the communication delays $t-\tau_{ij}(t)$ are independently generated from a uniform distribution on the set {1,2,...,B} and, for simplicity, we assume that $\mathcal{T}_i = \{1,2,\ldots\}$ for all i. [This models a situation where the computation delay is negligible compared to the communication delay.] The components of $x(1-B), x(2-B), \ldots, x(0)$ are independently generated from a uniform distribution over the interval [0,10] (this is to reflect a lack of coordination amongst processors) and the algorithm terminates at time t if $\max_{\tau,\tau'\in\{t-B,\ldots,t\}}\|x(\tau)-x(\tau')\| \le$

.001.

The termination time of PASYN, for different values of γ, B and n, is shown in Figures 8.1.a-8.1.c. In general, the rate of convergence of PASYN is the fastest for γ near 1 and for B small, corroborating our intuition. The termination time grows quite slowly with the size of the problem n but quite fast with decreasing γ. For γ near 1, the termination time grows roughly linearly with B (but not when γ is near 0).

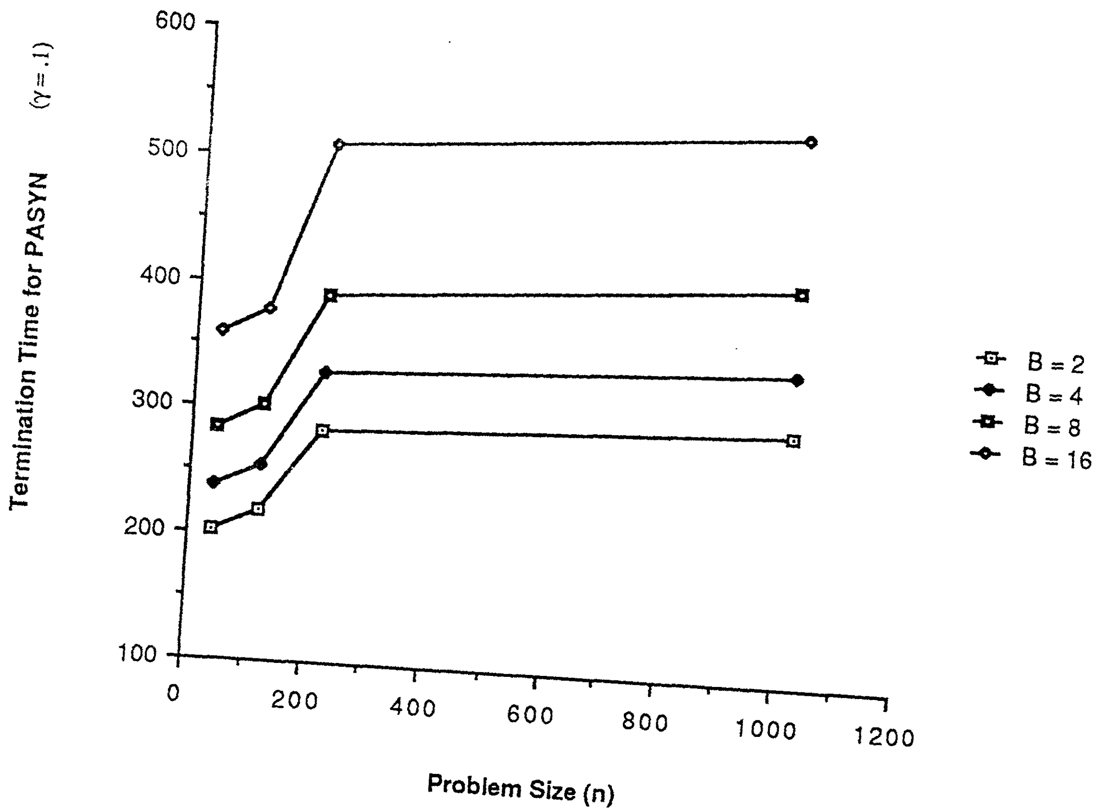


Figure 8.1.a. Termination time for PASYN (γ = .1), for different values of B and n.

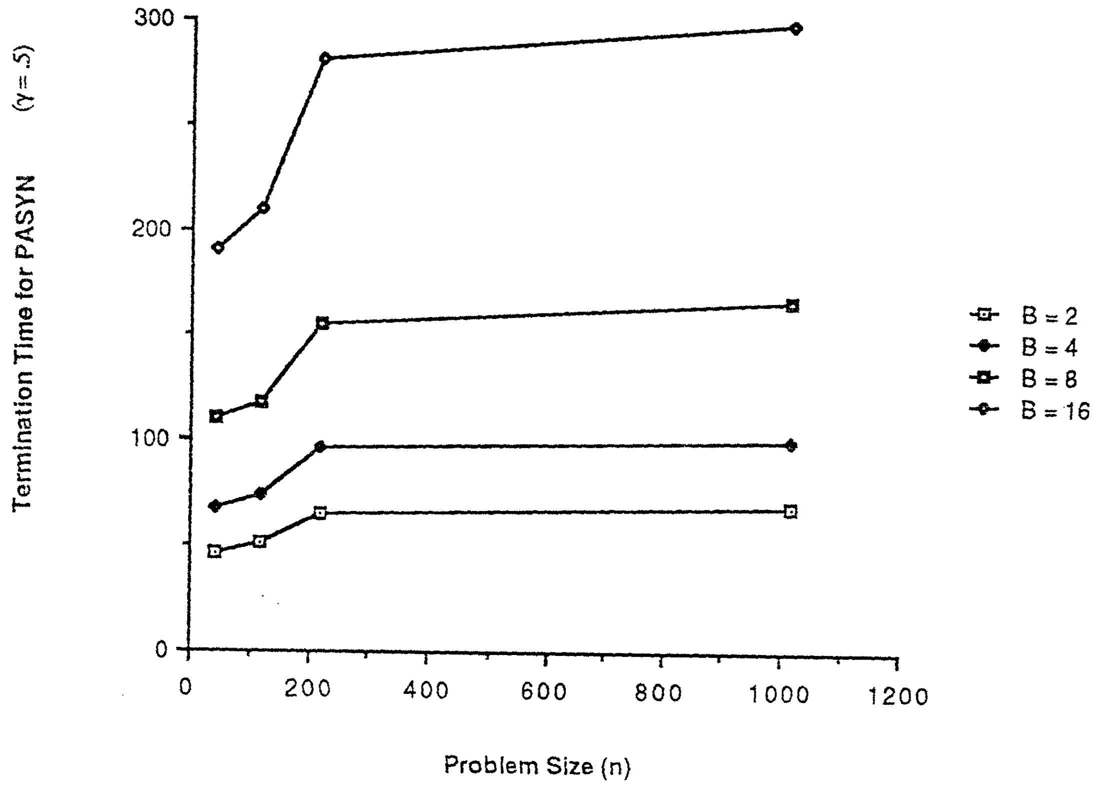


Figure 8.1.b. Termination time for PASYN (γ = .5), for different values of B and n.

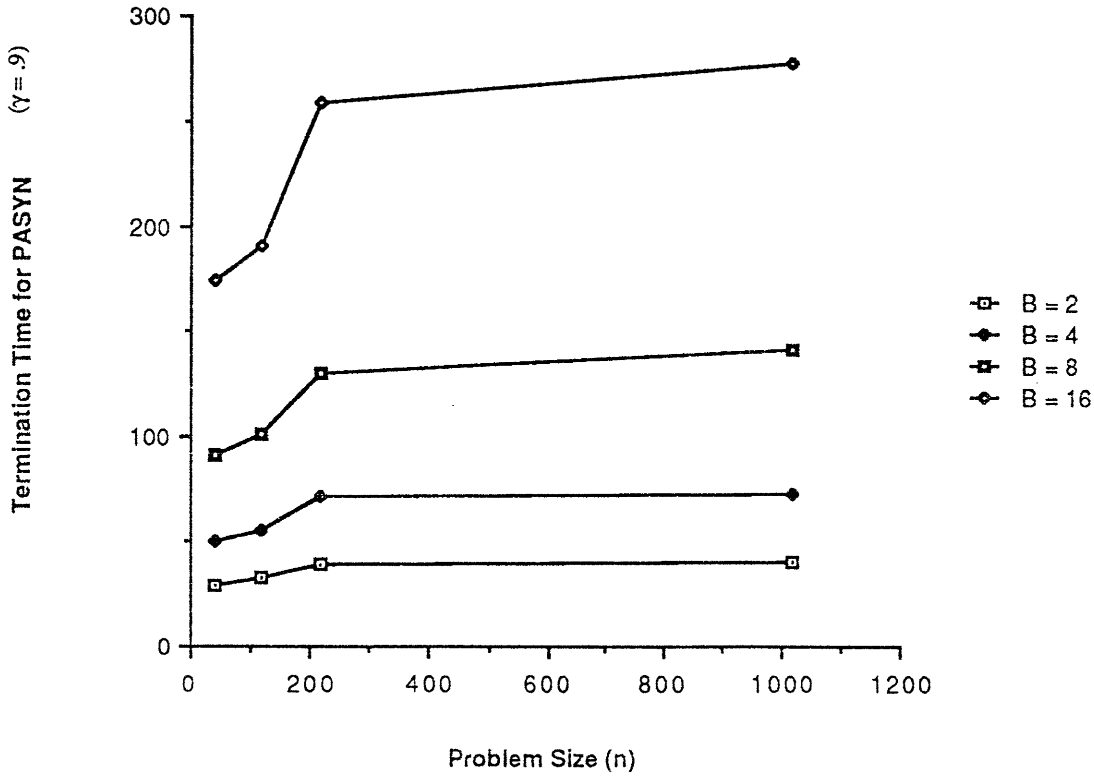


Figure 8.1.c. Termination time for PASYN (γ = .9), for different values of B and n.

8.3. An Alternative Partially Asynchronous Algorithm

Consider the function $f^{\circ}:\Re^n \rightarrow \Re^n$ whose ith component is given by

$$f_i^{\circ}(x) = \begin{cases} h_i(x) & \text{if } i \neq 1, \\ x_1 & \text{otherwise.} \end{cases} \tag{8.3}$$

It is shown in [1] that the algorithm $x := f^{\circ}(x)$ converges under the total asynchronism assumption. Hence it is of interest to compare this algorithm with that described in §8.2 (namely PASYN) under the same assumption of partial asynchronism. The partially asynchronous version of the algorithm $x := f^{\circ}(x)$, called TASYN, is identical to PASYN except that the function f in (8.2) is replaced by f°. [Note that TASYN has the advantage that it uses a unity stepsize.]

The termination time of TASYN, for different values of B and n, is shown in Figure 8.2. A comparison with Figures 8.1.a-8.1.c shows that TASYN is considerably slower than PASYN. The speed of TASYN is improved if f in (8.2) is replaced by f° only after a certain amount of time has elapsed, but the improvement is still not sufficient for it to compete with PASYN.

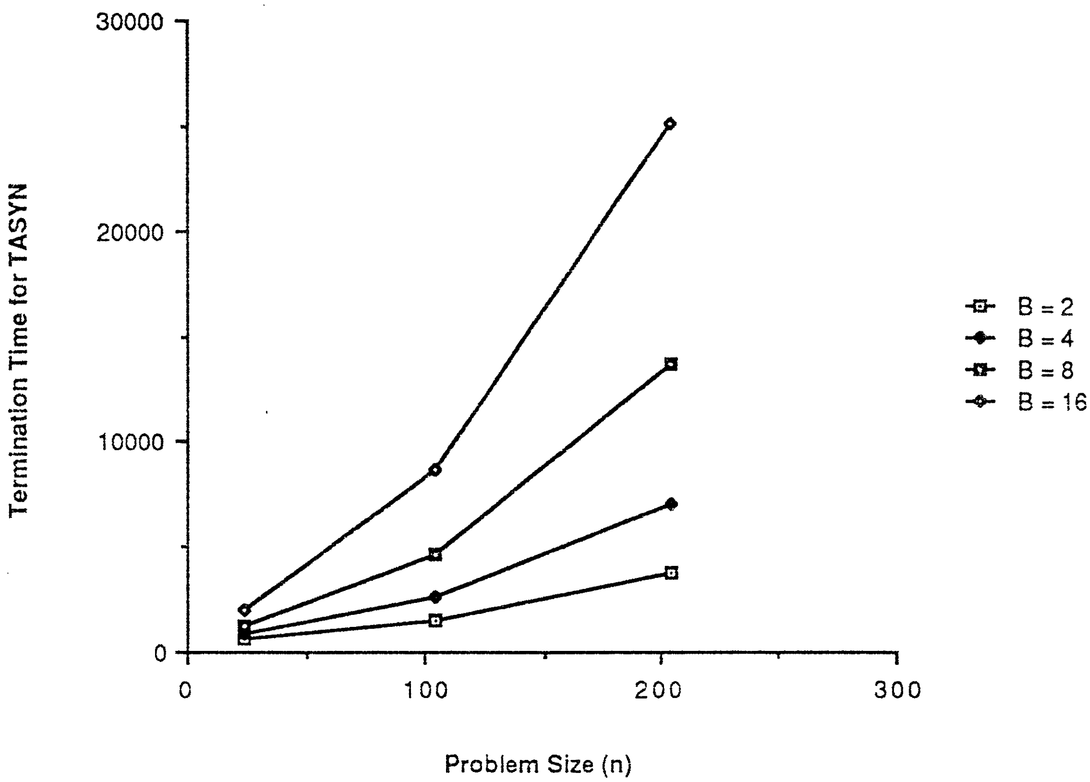


Figure 8.2. Termination time for TASYN, for different values of B and n.

8.4. Two Synchronous Algorithms

In this subsection we consider two types of synchronous algorithms based on h: the Jacobi algorithm and the Gauss-Seidel algorithm. In particular, the Gauss-Seidel algorithm been shown to be efficient for practical computation (see [22]-[24]). Hence, by comparing the asynchronous algorithms with these algorithms, we can better measure the practical efficiency of the former.

The Jacobi algorithm, called SYNJB, is a parallel algorithm that generates a sequence $\{x(t)\}$ according to

$$x(t+1) = (1-\gamma)x(t) + \gamma h(x(t)),$$

where $\gamma \in (0,1)$. The initial estimates $x_1(0), \ldots, x_n(0)$ are independently generated from a uniform distribution over the interval [0,10], and the algorithm terminates at time t if $\|x(t)-x(t-1)\| \leq .001$. [SYNJB can be seen to be a special case of PASYNB where $B = 1$ and hence $\{x(t)\}$ converges to a fixed point of h.]

Consider any positive integer b and any function $\beta: N \to \{1,\ldots,b\}$ such that $h_i(x)$ does not depend on x_j if $\beta(i) = \beta(j)$. We associate with b and β a Gauss-Seidel algorithm that generates a sequence $\{x(t)\}$ according to

$$x_i(t+1) = \begin{cases} h_i(x_1(t),\ldots,x_n(t)) & \text{if } t \equiv \beta(i)-1 \pmod{b}, \\ x_i(t) & \text{otherwise.} \end{cases}$$

In our simulation, the initial estimates $x_1(0), \ldots, x_n(0)$ are independently generated from a uniform distribution over the interval [0,10] and the algorithm terminates at time t if $\max_{\tau,\tau' \in \{t-b,\ldots,t\}} \|x(\tau)-x(\tau')\| \leq .001$. [Convergence of $\{x(t)\}$ to a fixed point of h follows from Proposition 2.4 in [2]. Note that, similar to TASYN, this algorithm has the advantage of using a unity stepsize.] We consider both a serial and a parallel version of this algorithm (this is done by choosing b and β appropriately). SYNGS1 is the serial version which chooses b = n and $\beta(i) = i$ for all i. SYNGS2 is the parallel version which uses a colouring heuristic to find, for each problem, a choice of b and β for which b is small.

The termination time for SYNJB, SYNGS1 and SYNGS2, for different values of n, are shown in Figures 8.3.a-8.3.b. In Figure 8.3.a, the choice of b obtained by the colouring heuristic in SYNGS2 is also shown (in parentheses). In general, SYNJB is considerably faster than either of the two Gauss-Seidel algorithms SYNGS1 and SYNGS2 (however in SYNJB all processors must compute at all times). From Figure 8.3.b we see that, as n increases and the problems become more sparse, SYNGS2 (owing to its high parallelism) becomes much faster than the serial algorithm SYNGS1. [Notice that the time for SYNGS1 is approximated by the time for SYNGS2 multiplied by n/b, as expected.] Comparing Figure 8.3.a with Figure 8.1.c, we see that SYNJB is approximately 3/2 times faster than PASYN and that SYNGS2 is slower than PASYN unless PASYN suffers long delays.

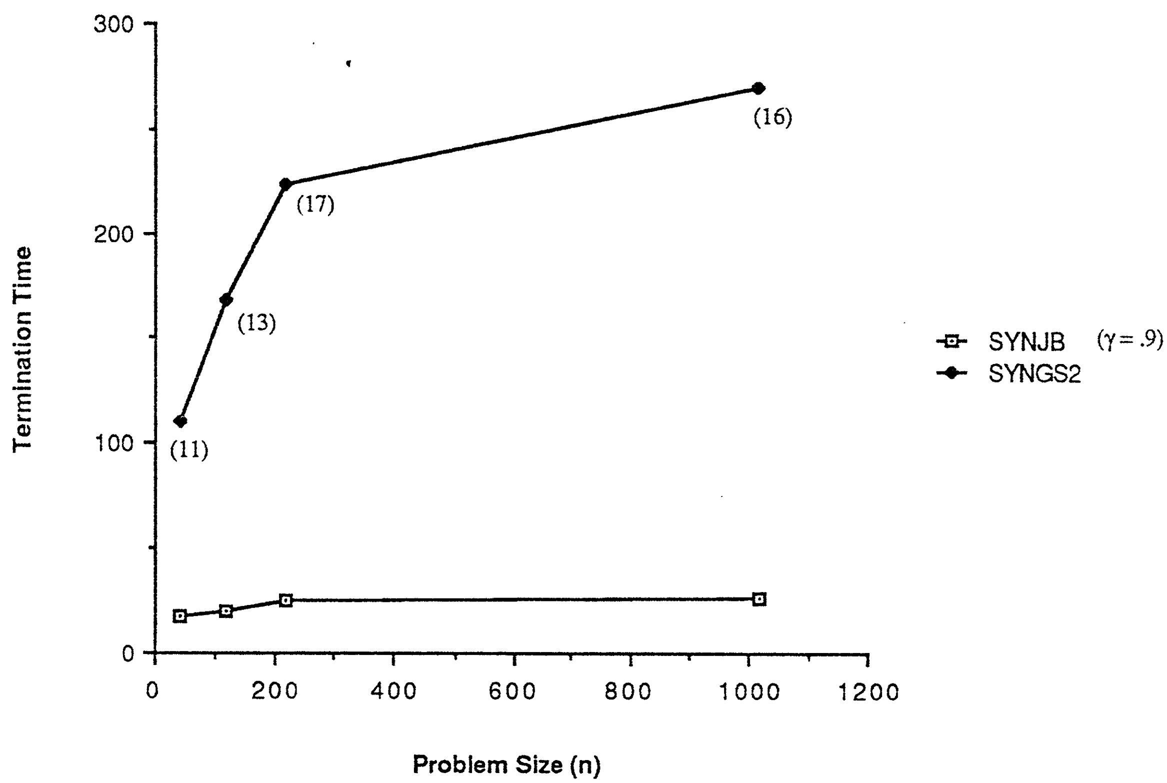


Figure 8.3.a. Comparing the termination time for the two synchronous, parallel algorithms SYNJB (γ=.9) and SYNGS2, for different values of n.

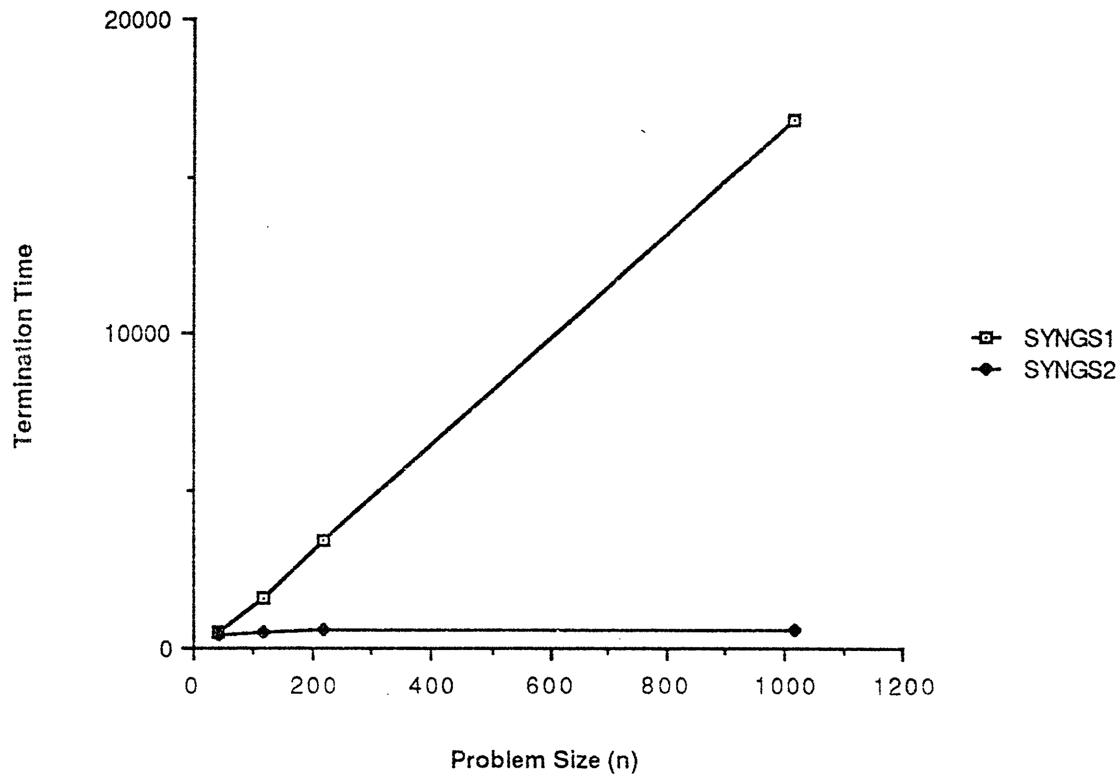


Figure 8.3.b. Comparing the termination time for the serial algorithm SYNGS1 and for the synchronous, parallel algorithm SYNGS2, for different values of n.

8.5. Simulation of Synchronous Algorithms in the Face of Communication Delays

In this subsection we consider the execution of the synchronous iterations of Subsection 8.4 in an asynchronous computing environment, that is, in an environment where communication delays are variable and unpredictable. The mathematical description of the algorithms in this subsection is identical to that of the algorithms considered in the preceding subsection; for this reason, the number of iterations until termination is also the same. On the other hand, each processor must wait until it receives the updates of the other processors before it can proceed to the next iteration. For this reason, the actual time until termination is different than the number of iterations. In our simulation, the delays are randomly generated but their statistics are the same as in our simulation of asynchronous algorithms in Subsections 8.2 and 8.3 (uniformly distributed over the set $\{1,\ldots,B\}$, where B denotes the maximum delay). This will allow us to determine whether asynchronous methods are preferable in the face of communication delays.

More precisely, consider any synchronous algorithm and let T denote the number of iterations at which this algorithm terminates. With each $t\in\{1,\ldots,T\}$ and each $i\in N$, we associate a positive integer $\sigma_i(t)$ to represent the "time" at which the update of the ith component at iteration t is performed in the corresponding asynchronous execution. [Here we distinguish between "iteration" for the synchronous algorithm and "time" for the asynchronous execution.] Then $\{\sigma_i(t)\}$ is recursively defined by the following formula

$$\sigma_i(t) = \max\{\ \sigma_j(t-1) + (\text{communication delay from proc. } j \text{ to proc. } i \text{ at time } \sigma_j(t-1))\ \},$$

where the maximization is taken over all $j\in N$ such that the jth component influences the ith component at iteration t. The termination time of the asynchronous algorithm is then taken to be

$$\max_{i\in N}\{\ \sigma_i(T)\ \}.$$

The partially asynchronous algorithms that simulate SYNJB, SYNGS1 and SYNGS2 are called, respectively, PASYNJB, PASYNGS1 and PASYNGS2. The termination times for these algorithms are shown in Figure 8.4-8.6 (they are obtained from the termination time shown in Figures 8.3.a-8.3.b using the procedure described above). Comparing these figures with Figures 8.1.a-8.1.c, we see that PASYNJB is roughly 4/3 times slower than PASYN (when both use the same stepsize γ = .9) while the other two algorithms PASYNGS1 and PASYNGS2 are considerably slower than PASYN (even when PASYN uses the most conservative stepsize γ = .1).

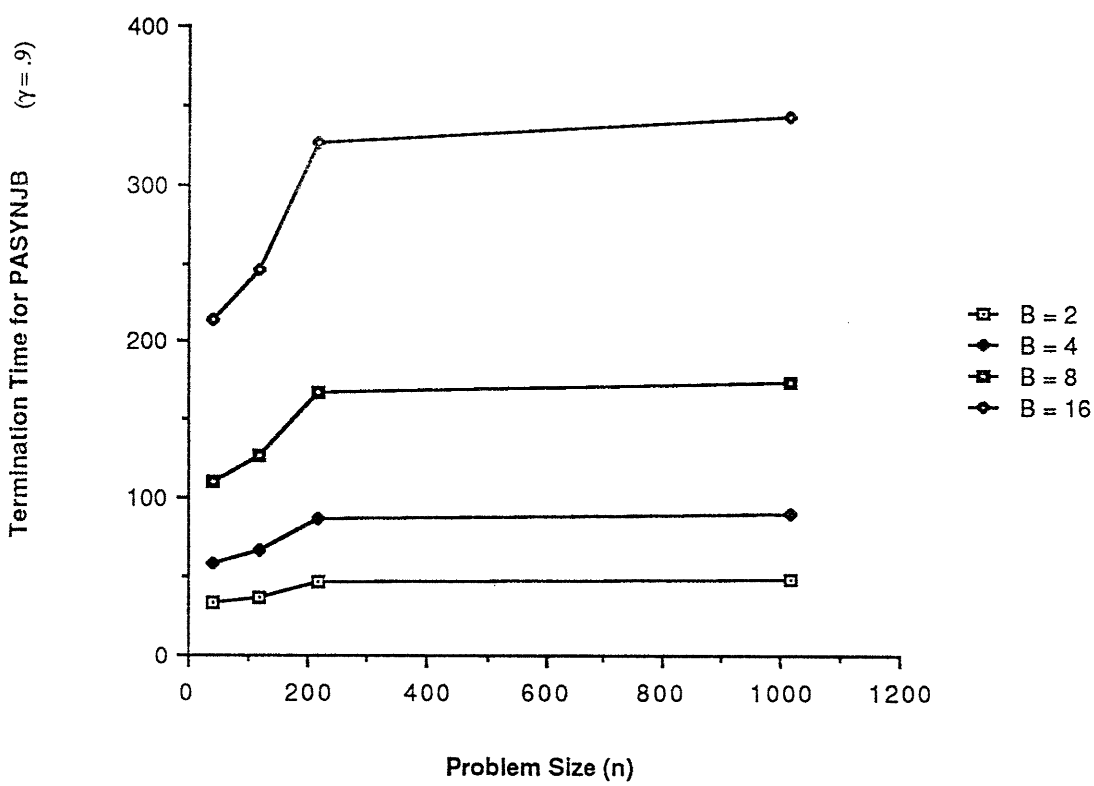


Figure 8.4. Termination time for PASYNJB (γ= .9), for different values of B and n.

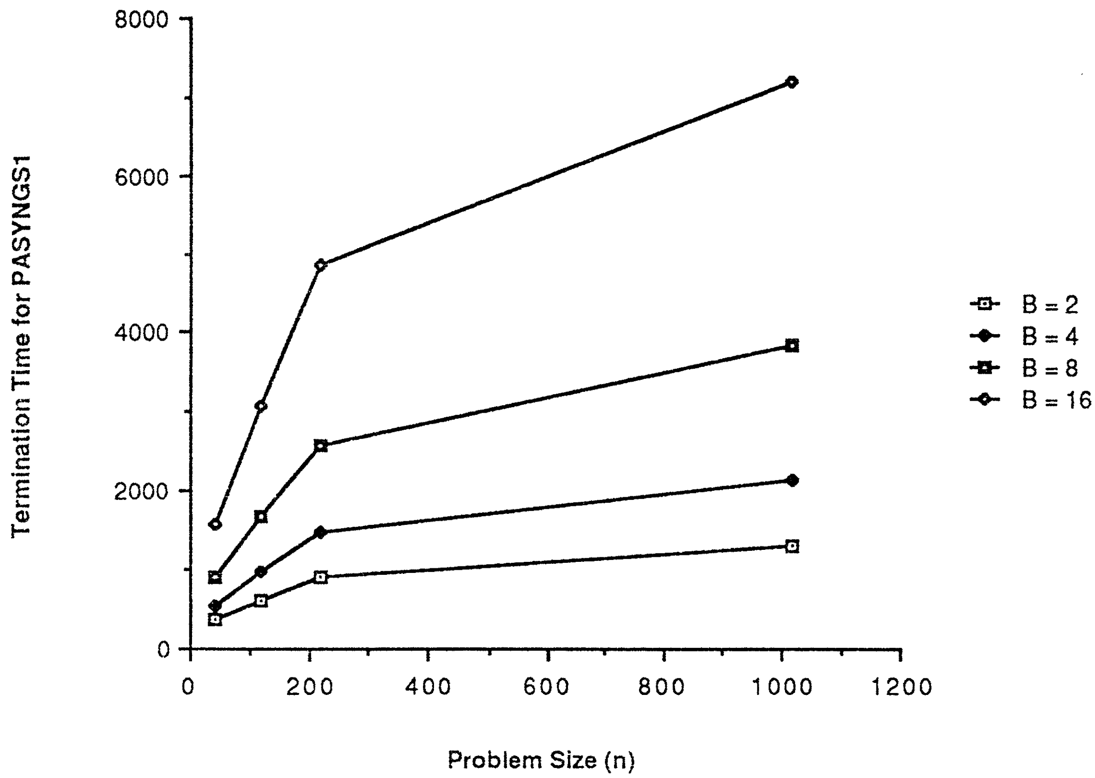


Figure 8.5. Termination time for PASYNGS1, for different values of B and n.

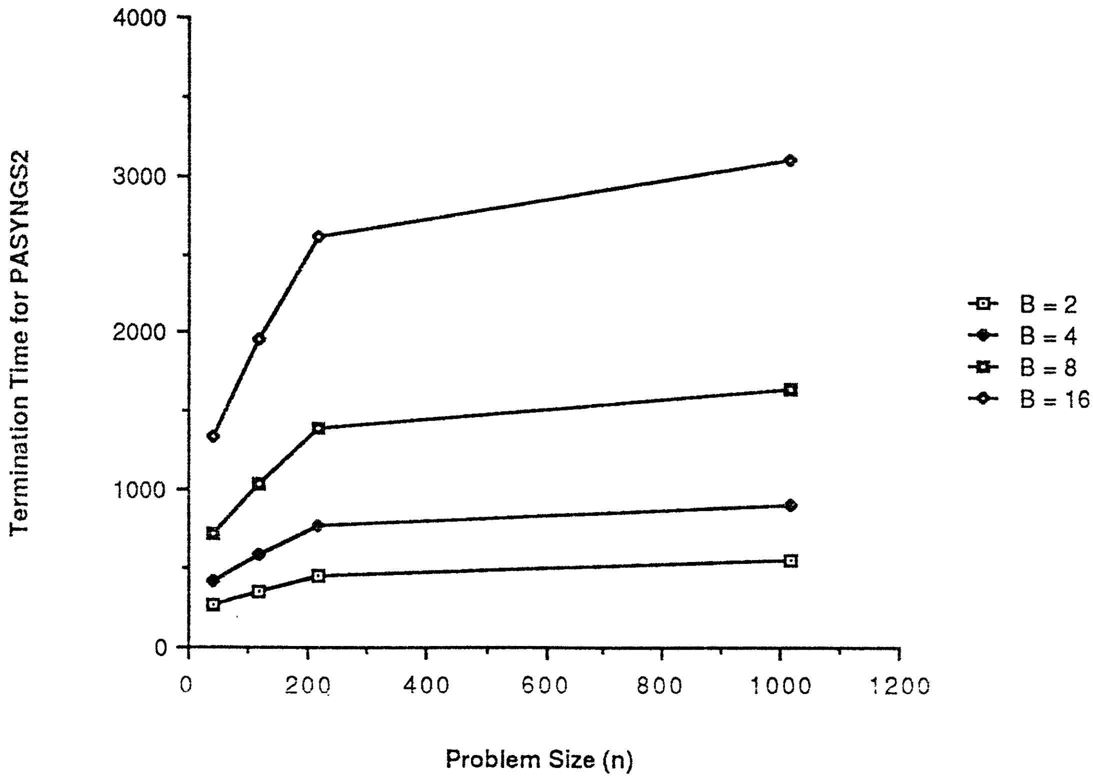


Figure 8.6. Termination time for PASYNGS2, for different values of B and n.

To summarize, we can conclude that PASYN is the fastest for partially asynchronous computation and that its synchronous counterpart SYNJB is the fastest for synchronous parallel computation. We remark that similar behaviour was observed in other network flow problems that were generated. Furthermore, the asynchronous algorithm PASYN seems to be preferable to its synchronous counterpart SYNJB in the face of delays. In practice, the assumption that the delays are independent and identically distributed might be violated. For example, queueing delays are usually dependent; also, the distance between a pair of processors who need to communicate could be variable, in which case the delays are not identically distributed. On the other hand, such issues cannot be simulated convincingly without having a particular parallel computing system in mind.

9. Conclusion and Extensions

In this paper we have presented a general framework, based on nonexpansive mappings, for partially asynchronous computation. The key to this framework is a new class of functions that are nonexpansive with respect to the maximum norm. We showed that any algorithm whose algorithmic mapping belongs to this class converges under the partial asynchronism assumption with an arbitrarily large bound on the delays. While some of the asynchronous algorithms thus obtained are known, others are quite new. Simulation with network flow problems suggests that the new algorithms may be substantially faster than the partially asynchronous implementation of serial algorithms. Whether these new algorithms are indeed competitive with other parallel algorithms or fast serial algorithms cannot be determined conclusively without further study.

In the future we hope to implement some of these algorithms on parallel computers to test their practical efficiency. In this direction, the work of [26] has shown much promise. It would also be of interest to expand the class of problems that come under our framework or to sharpen the convergence theory (by weakening our assumptions).

References

[1] D.P. Bertsekas and D. El Baz, "Distributed Asynchronous Relaxation Methods for Convex Network Flow Problems," SIAM J. Control and Optimization, 25 (1987), 74-85.

[2] D.P. Bertsekas, P. Hosein, and P. Tseng, "Relaxation Methods for Network Flow Problems with Convex Arc Costs," SIAM J. Control and Optimization, 25 (1987), 1219-1243.

[3] Bertsekas, D.P. and Tsitsiklis, J.N., Parallel and Distributed Algorithms: Numerical Methods, Prentice-Hall, N.J. (1988).

[4] D.P. Bertsekas, Dynamic Programming: Deterministic and Stochastic Models, Prentice-Hall, N.J. (1987).

[5] Tseng, P., "Distributed Computation for Linear Programming Problems Satisfying Certain Diagonal Dominance Condition," Working Paper 1256, Faculty of Commerce and Business Administration, U.B.C., also LIDS Report, M.I.T. (December 1986).

[6] Deimling, K., Nonlinear Functional Analysis, Springer-Verlag, Berlin (1985).

[7] Cottle, R.W. and Veinott, Jr., A.F., "Polyhedral Sets Having a Least Element," Math. Prog., 3 (1972), 238-249.

[8] Gabbay, H., "A Note on Polyhedral Sets Having a Least Element," Math. Prog., 11 (1976), 94-96.

[9] Ortega, J.M. and Rheinboldt, W.C., Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, N.Y. (1970).

[10] Rockafellar, R.T., Convex Analysis, Princeton University Press, N.J. (1970).

[11] Rockafellar, R.T., Network Flows and Monotropic Optimization, Wiley-Interscience, N.Y. (1984).

[12] Lubachevsky, B. and Mitra, D., "A Chaotic, Asynchronous Algorithm for Computing the Fixed Point of a Nonnegative Matrix of Unit Spectral Radius," J. Assoc. Comput. Mach., 33 (1986), 130-150.

[13] Tsitsiklis, J.N. and Bertsekas, D.P., "Distributed Asynchronous Optimal Routing in Data Networks," IEEE Trans. on Aut. Control, AC-31 (1986), 325-332.

[14] Tsitsiklis, J.N., Bertsekas, D.P. and Athans, M., "Distributed Asynchronous Deterministic and Stochastic Gradient Optimization Algorithms," IEEE Trans. on Aut. Control, AC-31 (1986), 803-812.

[15] Tsitsiklis, J.N., "Problems in Decentralized Decision Making and Computation", Ph.D. Thesis, Dept. of Electrical Engineering and Computer Science, MIT, Cambridge, MA (1984).

[16] Chazan, D. and Miranker, W., "Chaotic Relaxation," Linear Algebra and Its Applications, 2 (1969), 199-222.

[17] Baudet, G.M., "Asynchronous Iterative Methods for Multiprocessors," J. Assoc. Comput. Mach., 15 (1978), 226-244.

[18] Bertsekas, D.P., "Distributed Dynamic Programming," IEEE Trans. Auto. Contr., AC-27 (1982), 610-616.

[19] Bertsekas, D.P., "Distributed Asynchronous Computation of Fixed Points," Math. Prog., 27 (1983), 107-120.

[20] Tsitsiklis, J.N., "On the Stability of Asynchronous Iterative Processes," Mathematical Systems Theory, 20 (1987), 137-153.

[21] Klingman, D., Napier, A. and Stutz, J., "NETGEN-A Program for Generating Large Scale (Un)capacitated Assignment, Transportation and Minimum Cost Flow Network Problems," Mgmt. Sci., 20 (1974), 814-822.

[22] Ohuchi, A. and Kaji, I., "Lagrangian Dual Coordinatewise Maximization Algorithm for Network Transportation Problems with

Quadratic Costs," Networks, 14 (1984), 515-530.

[23] Cottle, R.W., Duvall, S.G. and Zikan, K., "A Lagrangean Relaxation Algorithm for the Constrained Matrix Problem," Naval Research Logistics Quarterly, 33 (1986), 55-76.

[24] Zenios, S.A. and Mulvey, J.M., "Relaxation Techniques for Strictly Convex Network Problems," Annals of Operations Research 5: Algorithms and Software for Optimization (Monma, C.L. ed.), Scientific Publishing Company, Switzerland (1986).

[25] Dembo, R.S. and Klincewicz, "A Scaled Reduced Gradient Algorithm for Network Flow Problems with Convex Separable Costs," Mathematical Programming Study, 15 (1981), 125-147.

[26] Zenios, S.A. and Mulvey, J.M., "A Distributed Algorithm for Convex Network Optimization Problems," Parallel Computing, 6 (1988), 45-56.

[27] James, K. R., "Convergence of Matrix Iterations Subject to Diagonal Dominance," SIAM J. Numer. Anal., 10 (1973), 478-484.

[28] Lippmann, R.P., "An Introduction to Computing with Neural Nets," IEEE ASSP Magazine, (April, 1987), 4-22.

[29] Hopfield, J.J., "Neurons with Graded Response Have Collective Computational Properties Like Those of Two-State Neurons," Proc. Natl. Acad. Sci. USA, 81 (1984), 3088-3092.

[30] Hopfield, J.J. and Tank, D.W., "Computing with Neural Circuits: A Model," Science, 233 (1986), 625-633.

[31] Gantmacher, F. R., The Theory of Matrices, Vol. 2, Chelsea, N. Y. (1960).